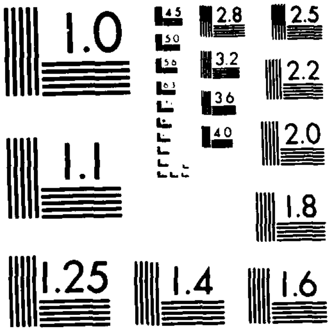
AD-A174 998   RHEOLOGY OF DISSEMINATION(U) RHEOLOGY RESEARCH   1/1
GAITHERSBURG MD E A KEARSLEY OCT 86 CRDEC-CR-87022
DAAK11-83-C-0040

UNCLASSIFIED   F/G 20/4   NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

CHEMICAL
RESEARCH,
DEVELOPMENT &
ENGINEERING
CENTER

CRDEC-CR-87022

RHEOLOGY OF DISSEMINATION
PHASE II

by E. A. Kearsley
RHEOLOGY RESEARCH
Gaithersburg, MD 20879

October 1986

U.S. ARMY
ARMAMENT
MUNITIONS
CHEMICAL COMMAND

Aberdeen Proving Ground, Maryland 21010-5423

Disclaimer

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorizing documents.

Distribution Statement

Approved for public release; distribution is unlimited.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

AD-A174 998

REPORT DOCUMENTATION PAGE

1a REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

1b RESTRICTIVE MARKINGS

2a SECURITY CLASSIFICATION AUTHORITY

2b DECLASSIFICATION / DOWNGRADING SCHEDULE

3 DISTRIBUTION / AVAILABILITY OF REPORT
Approved for public release; distribution is unlimited.

4 PERFORMING ORGANIZATION REPORT NUMBER(S)
CRDEC-CR-87022

5 MONITORING ORGANIZATION REPORT NUMBER(S)

6a NAME OF PERFORMING ORGANIZATION
Rheology Research

6b OFFICE SYMBOL *(If applicable)*

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS *(City, State, and ZIP Code)*
P.O. Box 2101
Montgomery Village
Gaithersburg, MD 20879

7b. ADDRESS *(City, State, and ZIP Code)*

8a. NAME OF FUNDING / SPONSORING ORGANIZATION
CRDEC

8b. OFFICE SYMBOL *(If applicable)*
SMCCR-RSP-P

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
DAAK-11-83-C-0040

8c ADDRESS *(City, State, and ZIP Code)*
Aberdeen Proving Ground, MD 21010-5423

10 SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.

11 TITLE *(Include Security Classification)*
Rheology of Dissemination. Phase II

12 PERSONAL AUTHOR(S)
Kearsley, E. A.

13a. TYPE OF REPORT
Contractor

13b. TIME COVERED
FROM 84 Jul TO 86 Jan

14. DATE OF REPORT *(Year, Month, Day)*
1986 October

15. PAGE COUNT
58

16 SUPPLEMENTARY NOTATION
COR: Joseph Matta, Ph.D., SMCCR-RSP-P, (301) 671-4285

17 COSATI CODES

FIELD	GROUP	SUB-GROUP
15	02	

18. SUBJECT TERMS *(Continue on reverse if necessary and identify by block number)*
BKZ model, Finite differences, Rheology, Drop formation, Liquid jets, Satellite drops, Filament breakup, Numerical techniques, Surface tension

19 ABSTRACT *(Continue on reverse if necessary and identify by block number)*

The breakup of a cylindrical jet of BKZ fluid perturbed sinusoidally is modeled in one dimension. Surface tension, nonlinear viscoelastic forces and both radial and axial acceleration are taken into account. The model leads to an integro-differential equation. A FORTRAN program is developed to solve this equation using a predictor-corrector method. Various subroutines are given to evaluate the necessary derivatives with finite differences and to solve a pentadiagonal matrix which arises. The techniques and routines developed are useful for solving a variety of similar problems.

20 DISTRIBUTION / AVAILABILITY OF ABSTRACT
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION
UNCLASSIFIED

22a NAME OF RESPONSIBLE INDIVIDUAL
TIMOTHY E. HAMPTON

22b TELEPHONE *(Include Area Code)*
(301) 671-2914

22c. OFFICE SYMBOL
SMCCR-SPS-T

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

PREFACE

The work described in this report was authorized under Contract No. DAAK-11-83-C-0040. This work was started in July 1984 and completed in January 1986.

The use of trade names or manufacturers' names in this report does not constitute an official endorsement of any commercial products. This report may not be cited for purposes of advertisement.

Reproduction of this document in whole or in part is prohibited except with permission of the Commander, U.S. Army Chemical Research, Development and Engineering Center, ATTN: SMCCR-SPS-T, Aberdeen Proving Ground, Maryland 21010-5423. However, the Defense Technical Information Center and the National Technical Information Service are authorized to reproduce the document for U.S. Government purposes.

This report has been approved for release to the public.

Blank

CONTENTS

Blank

RHEOLOGY OF DISSEMINATION. PHASE II

1. INTRODUCTION

The project Rheology of Dissemination, contract number DAAK 11-83-C-0040, is developing the rheology of polymer solutions necessary for understanding the processes of breakup of these liquids in airstreams. In the period July 1984 through December 1985 the activities of this project have centered on setting up computational techniques for doing numerical studies of the breakup processes. The work of this period has been concerned with the derivation and solution of a particular integro-differential equation; the aim has been to develop techniques and Fortran subroutines useful in looking at a panoply of problems as well as to develop general insight and a feel for the importance of the various terms and physical parameters. A particular problem was chosen for this purpose, a relatively simple one with similarities to many of the common, practical methods of drop formation. The breakup of a cylindrical jet under sinusoidal perturbation is a well known and relatively simple phenomenon which probably exhibits the important features of more complex drop forming processes. It has offered more than enough complexities to fully occupy our energies and the resulting algorithms and the experience in applying them seem well worth the effort. Since the benefits are expected to apply further and to be useful to other researchers, the numerical program has been constructed modularly and is well annotated to make it readily available and to allow the various sections and subroutines to be extracted and used separately.

The breakup into droplets of a jet of liquid in air is a topic of long standing. In the early nineteenth century Savart [1833] initiated an experimental study of this topic and Plateau [1856] established that the effect was dominated by surface tension forces. (Curiously, though his meaning is unmistakeable, Plateau never uses the term "surface tension"; perhaps it had not yet been coined). Some years thereafter Lord Rayleigh [1879] was drawn to the problem by an interest in so-called "sensitive" jets and flames which respond strongly to low level acoustic disturbances. Rayleigh's simple linearized analysis of the stability of an inviscid liquid jet established the importance of the wavelength of a disturbance and showed that only those disturbances would grow that were axisymmetric and with wavelengths greater than the diameter of the jet. Subsequent experiments in this field have looked at drop formation of jets, usually subjected to a small periodic disturbance of known frequency.

Over the last thirty years there has been a resurgence of interest in certain aspects of the problem, first because of a possible connection with a troublesome combustion instability in liquid rocket engines, then as an important problem for the development of ink-jet printers. These recent studies have focussed on the breakup of a filament of Newtonian liquid issuing in a steady flow from a nozzle with a superimposed sinusoidal fluctuation. Breakup begins by the formation of a string of connected ellipsoidal drops a wavelength apart. The ligament of fluid connecting adjacent drops may consolidate with them upon final breakup or, under different conditions, may form a separate, so-called "satellite" drop. Some striking photographs of this process are to be found in the literature (e.g. Donnelly and Glaberson [1966]). Given the observation that as drop formation progresses the wavelength is preserved, it is easy to conclude that the drops would each be equal to the volume of a wavelength of the original cylindrical jet were it not for the possibility of formation of satellite droplets. The suppression of these satellite droplets is of great importance in the design of ink-jet printers.

It was recognized early in these studies that the formation of droplets is inherently a nonlinear process. In an effort to take this into account the stability analysis of Rayleigh has been generalized by carrying the perturbation terms up to third order (Lafrance [1975]). Taub [1976] has devised a clever optical measuring device which he used to detect these high order terms in the early stages of the process. There are few attempts to solve the full Navier Stokes equations in this geometry. Commonly, some form of restricted one-dimensional approximation is assumed which neglects troublesome terms. Recent numerical studies often handle the nonlinearities more completely by using a one dimensional formulation by Bogy [1977] based on the results of Green [1976] who applies Cosserat theory to the problem.

The observation that a small amount of high molecular weight polymer dissolved in the liquid of the jet has a marked effect on the breakup process has suggested further applications and aroused even more interest in the problem. Demisting of aircraft fuels and the atomization of fuels and other materials for dispersion can be controlled in this way. A new element is added to the system because the elastic forces induced by the polymer have a remarkable stabilizing effect on the breakup process. Experiments of Gordon, Yerushalim and Shinnar [1973] show that with solutions of Separan and other polymers the drops form regularly at the wavelength of the disturbance just as with Newtonian liquids but that the ligaments between drops persist for a very much longer time. Consequently, the formation of satellite drops is strongly influenced. Some striking photographs illustrating the phenomenon are found in the paper mentioned.

Keunings [1984, 1986] has published a series of papers on a two dimensional finite element calculation of this problem including nonlinear viscoelastic effects. His method accomodates "rate-type" models of liquids, that is, liquids for which the viscoelastic stresses can be calculated from instantaneous kinematic quantities. He compares the results for a Newtonian liquid and an Oldroyd-B fluid (a generalization of the Maxwell model). Without serious modifications Keunings' method does not seem suitable for dealing with materials which require that the history of deformation be used for calculation of the stress. Furthermore, the methods requires considerable computation time and the steady hand of an experienced numerical analyst. Bousfield et al. [1984a,1984b,1986] have developed a one dimensional system for rate type models and have compared their results with those of Keunings. Their calculations for both Newtonian and Maxwell model liquids, when they do not encounter convergence problems, are in good agreement with the more complete calculations of Keunings. Their equations do not include radial inertia effects.

The Maxwell model is a member of the class of materials which, for sufficiently smooth histories, may be viewed as either of rate type or of BKZ type. Therefore, Keunings' results for an Oldroyd-B model can be used to test the validity of the method developed here, a one dimensional approximation scheme designed to handle BKZ materials. Similarly, the results of Bousfield et al. can also be compared directly.

2. DESCRIPTION OF PROBLEM AND LAYOUT OF REPORT

Consider an infinite cylindrical filament of liquid at rest and not acted on by gravity or otherwise accelerated. An internal pressure will balance the forces of surface tension and the cylinder will be in equilibrium. This equilibrium is not an energy minimum, however, and is inherently unstable. If at time t=0 we suddenly perturb the filament with a small, sinusoidal displacement along its axis while setting the velocity and acceleration again equal to zero, the disturbance will grow in time under the influence of surface tension and viscoelastic forces. The periodicity of the deformation will be preserved as the disturbance develops, droplets will form regularly along the axis of the cylinder and ultimately they will separate. It was our aim to model this process for a liquid of the BKZ type for which the viscoelastic stresses depend upon the history of the deformation in the neighborhood of a point. The works cited for Keunings and for Bousfield et al also model this process but for rate-type viscoelastic materials.

We begin by modeling the various forces acting on the filament; surface tension, viscoelastic forces and a pressure due to radial inertia. These effects are presented in the three following sections. In the sixth section an integrodifferential equation is formed by equating the gradient of the sum of these forces to the local axial acceleration of the cylinder. This equation and the boundary and initial conditions are nondimensionalized. In section seven some scaling and conditioning to prepare for the numerical treatment of the equation and boundary conditions are explained . In the eighth section a description of the algorithm for solving the equation is presented, including some details of the method used for numerical integration in calculating the viscoelastic force gradient. Section nine outlines the mechanics of using the Fortran program, DROPGEN, listing the parameters that must be supplied and explaining the form of the output. Section ten is a complete listing and a concise description of the quantities which are used in the program. Section eleven presents the actual Fortran program DROPGEN and its subroutines, DERIV, SLOPES and VBAND and section thirteen displays an example of the printed output for a typical run of the program.

The first two appendices explain the Fortran subroutines; DERIV and SLOPES, using the central difference method for approximating derivatives, VBAND which solves a set of linear equations by the inversion of a pentadiagonal matrix. The third appendix explains the technique for extrapolating in time the kinematic quantities needed for the predictor/corrector technique. The final appendix sketches a method of calculating the viscoelastic force gradient which does not require the cumbersome storing of a complete history of the deformation and which is sometimes possible for certain types of BKZ materials.

3. SURFACE TENSION FORCES

The net effect of the surface tension on a cross section of the filament arises from two separate phenomena. There are surface tension forces acting directly on the circumference of the cross section and, in addition, the surface tension forces generate a pressure, P_{st}, which acts on the cross section. This pressure may be found by a method similar to that used by Avula (1973) for calculating the static shapes of liquid drops. Envision the surface of the filament ruled with circumferential and meridional coordinates and designate the meridional coordinate as xi. Then xi is related to the radius R and the axial position Z through the Pythagorean theorem on the differentials, i.e.:

$$dZ^2+dR^2= d\xi^2 \qquad (3.1)$$

Let K_1 and K_2 be the principal curvatures in the meridional and circumferential directions, respectively; then we can write

$$K_1=-R_{\xi\xi}/(1-R_{\xi}^2)^{-1/2} \qquad (3.2)$$

$$K_2=(1-R_{\xi}^2)^{1/2}/R \qquad (3.3)$$

The surface tension acts as an elastic membrane on the curved surface of the infinitesimal cylinder inducing a pressure, P_{st}, which acts on the ends. This pressure is given by the following expression:

$$\sigma(K_1+K_2)= P_{st} \qquad (3.4)$$

where σ is the surface tension (assumed constant and isotropic in the surface). Upon substituting for the curvatures from equations (2) and (3) into this expression, one obtains the following expression for the pressure due to surface tension:

$$P_{st}= \sigma(1-R_{\xi}^2-RR_{\xi\xi})/R(1-R_{\xi}^2)^{1/2} \qquad (3.5)$$

We will need an expression for P_{st} in terms of Z rather than xi. To change variables we use equation (3.1) to generate the transformation equations.

$$R_{\xi}=R_Z/(1+R_Z^2)^{1/2} \qquad (3.6)$$

$$R_{\xi\xi}=R_{ZZ}/(1+R_Z^2)^2 \qquad (3.7)$$

With these substitutions equation (3.5) transforms as follows:

$$P_{st}= \sigma(1+R_Z^2-RR_{ZZ})/R(1+R_Z^2)^{3/2} \qquad (3.8)$$

In general P_{st} is a function of Z but constant over a cross section.

The force on the cross section is then the sum of the direct surface tension forces acting on the edge and the pressure acting over the surface, that is:

$$Fst=2\pi R\sigma/(1+R_Z^2)^{1/2}-\pi R^2P_{st} \qquad (3.9)$$

which becomes upon substituting for P_{st} from equation (3.8):

$$Fst=\pi R\sigma(1+R_Z^2+RR_{ZZ})/(1+R_Z^2)^{3/2} \qquad (3.10)$$

It is instructive to demonstrate equation (3.8) for the case of a static spherical liquid drop of radius ρ under no external forces. In that case R is given by the equation

$$R^2+Z^2=\rho^2 \qquad (3.11)$$

and using this in equation (3.8) gives a constant pressure

$$P_{st}=2\sigma/\rho \qquad (3.12)$$

which is a well known result. Furthermore, the force on any cross-section of the static spherical drop is zero according to equation (3.10) for the geometry of equation (3.11). This also is as it should be.

4. VISCOELASTIC FORCES

The viscoelastic forces acting on a cross section of the filament arise from a stress induced by the history of deformation of the material points of the cross section. A strictly accurate history of deformation of a warping cross section requires huge quantities of data but, fortunately, this is not necessary for our purposes. To avoid the problem we assume that every cross section of the filament always remains plane and perpendicular to the axis while deforming. That is, we assume that within any cross section the stretching is identical for every point. Such a deformation is necessarily purely extensional. Of course this can not be true if there is any axial dependence of the deformation. In such a circumstance compatibility conditions will not be met. For our purposes, however, the approximation should be valid as long as the relative change in radius along the axis is small with respect to one.

For a BKZ material in simple extension, the stress is calculated from the history of stretching at each material point. We consider a particle of material at reference point Z_0 on the axis of the cylinder which is at point $Z(Z_0,t)$ at time t. The stretch relative to Z_0 is

$$\lambda(Z_0,t)=\frac{\partial Z}{\partial Z_0} \tag{4.1}$$

The relative stretch from time τ to t is then

$$\lambda(Z_0;\tau,t)=\frac{\lambda(Z_0,t)}{\lambda(Z_0,\tau)} \tag{4.2}$$

The Cauchy stress, T, (true stress as opposed to engineering or Piola stress) is given by an integral over the history of the stretching:

$$T(Z_0,t)=\int_{-\infty}^{t} M(\lambda(Z_0;\tau,t),t-\tau)\ d\tau \tag{4.3}$$

M is a constitutive function of the material. It is a function of the relative stretch and of the time interval from a time in the past, τ, to the present time, t. Since the stress is assumed constant over a cross section of the filament the net viscoelastic force acting on the cross section is given by

$$Fve=\pi R^2\ T=\pi R^2 \int_{-\infty}^{t} M(\lambda(Z_0;\tau,t),t-\tau)\ d\tau \tag{4.4}$$

For BKZ materials in general, M is expressible in terms of the stress relaxation function through its time derivative. If H is the stress relaxation function, we can write

$$\begin{aligned} &M(\lambda(Z_0;\tau,t),t-\tau)=-H_*(\lambda(t,\tau),t-\tau) \\ &H_*(\lambda,t)=\frac{d}{dt}H(\lambda,t) \end{aligned} \tag{4.5}$$

If we rewrite equation (4.4) in terms of the stress relaxation function, $H(\lambda,t)$, the equation takes on the following form:

$$\begin{aligned} &Fve=\pi R^2 H(\lambda(Z_0,t),t)- \\ &\quad -\pi R^2 \quad H_*(\lambda(Z_0;\tau,t),t-\tau)\ d\tau \end{aligned} \tag{4.6}$$

For our sample calculation we will use a particularly simple form of BKZ fluid, the Maxwell model. For this model we have

$$H(\lambda,t)= G_0(\lambda^2-1/\lambda)/\exp(\alpha t) \qquad (4.7)$$

G_0 plays the role of an infinitesimal shear modulus and α is the reciprocal of a relaxation time.

Of course, it is possible to use any other form of BKZ fluid simply by programming a suitable calculation of the stress gradient. The Maxwell model is particularly useful as a test case because one can compare results with the full two dimensional finite element calculations of Keuning et al.

5. INERTIAL FORCES DUE TO RADIAL MOTION

The radial accelerations of an infinitesimal length of the filament are determined by the axial stretching. The acceleration is not constant over a cross section, however, and it generates a radial pressure gradient. The axial variation of this pressure contributes to the axial force acting on sections of the filament.

The radius of filament at any point is constrained by the stretching on the axis and is given by the equation

$$R(Z_0,t)=R_0/(\lambda(Z_0,t))^{1/2} \qquad (5.1)$$

where R_0 is the radius of the unstretched filament and $\lambda(Z_0,t)$ is the stretch which is given by equation (4.1). This equation is a consequence of the isochoric deformation imposed by incompressibility. A similar equation holds for the radius of an arbitrary point on a cross section, i.e.:

$$r(Z_0,t)=r_0/(\lambda(Z_0,t))^{1/2} \qquad (5.2)$$

where r and r_0 have the obvious significance. The velocity and acceleration in the radial direction can be calculated from this equation by taking derivatives with respect to time. In the following equations time derivatives are indicated by superposing a dot.

$$\dot{r} = -(r\dot{\lambda}/2\lambda) \qquad (5.3)$$

$$\ddot{r} = r(3\dot{\lambda}^2-2\lambda\ddot{\lambda})/4\lambda^2 \qquad (5.4)$$

This acceleration generates a radial variation of the pressure. If one calculates the the mass times the radial acceleration of an element of material at radius r and equates it to the net radial force on the element, one arrives at the following ordinary differential equation for the pressure:

$$\frac{d}{dr}(rP_{ri}(r))-P_{ri}(r)=\rho r^2(2\lambda\ddot{\lambda}-3\dot{\lambda}^2)/4\lambda^2 \qquad (5.5)$$

The solution of this equation can be adjusted to contribute zero pressure on the surface of the filament by adding an arbitrary pressure constant over the cross section. The arbitrary pressure will include the pressure due to surface tension and the isotropic part of the viscoelastic stress, both of which have been calculated seperately and both of which are constant over a cross section. The contribution of radial inertia to pressure is then given by:

$$P_{ri}(r)=\rho(2\lambda\ddot{\lambda}-3\dot{\lambda}^2)(r^2-R^2)/8\lambda^2 \qquad (5.6)$$

A contribution to the force acting on a cross section of the filament is found by integrating this pressure over the area of the cross section. The force in the positive Z direction exerted by the material on the positive side is given by:

$$F_{ri}=\pi\rho R^4(2\lambda\ddot{\lambda}-3\dot{\lambda}^2)/16\lambda^2 \qquad (5.7)$$

This force may be expressed in terms of the radius of the filament through use of equation (5.1) to replace the time derivatives of the stretch with those of the radius. When this is done, one arrives at the following simple form:

$$F_{ri}(Z_0,t)=-\pi\rho R^3 \ddot{R}/4 \qquad (5.8)$$

This result may be compared with a calculation of Green [1976] in which, by means of a Cosserat model with two directors, he formulates equations for a one-dimensional, straight jet of Newtonian fluid. Green's results are expressed in laboratory coordinates rather than the material coordinates we are using here, so that the comparison is not immediate. The term of Green's equations which is attributable to the effects of radial inertia has been isolated by Bogy [1978]* and can be written as follows:

$$F_{ri}(Z,t)=\pi\rho R^4(U_{Zt}+UU_{ZZ}-\frac{1}{2}U_Z^2)/8 \qquad (5.9)$$

where U is the axial velocity of the cylinder expressed in laboratory coordinates and the subscripts Z and t indicate differentiation. This equation can be expressed in material coordinates by means of the following transformation:

$$\begin{aligned} &U(Z,t)=V(Z_0,t)=\dot{Z}(Z_0,t) \\ &U_t=V-R^2VV_{Z_0} \qquad (5.10) \\ &U_Z=R^2V_{Z_0}\text{, etc.} \end{aligned}$$

It is extremely gratifying that equations (5.9), when expressed in material coordinates, is identical to equation (5.8).

* There is a misprint resulting in the dropping of a factor v in the fourth term of equation (8) of Bogy [1978] which, if unnoticed, can be very troublesome.

6. FORMULATION OF THE BOUNDARY VALUE PROBLEM

6.1 The Differential Equation.

We can write a differential equation describing the dynamics of the motion of the filament by collecting together the forces acting along the axis on a cross section of the filament. We ignore the force of gravity and all other body forces, although they can easily be included. The following equation results:

$$\frac{\partial}{\partial z}(F_{ve}+F_{st}+F_{ri})=\pi\rho R^{2}\ddot{Z} \qquad (6.1)$$

where the right hand side of this equation is the mass per unit length times the acceleration of point Z on the axis of the cylinder and the left hand side is the Z derivative of the net force in the axial direction (positive toward increasing Z) on a cross section of the filament. The equations for the various axial forces developed in the three previous sections are to be substituted into the left-hand side of this equation. When we have done so, the quantity pi may be factored out and we obtain the following equation:

$$\frac{\partial}{\partial z}(R^{2}T)+\sigma\frac{\partial}{\partial z}(R(1+R_{Z}^{2}+RR_{ZZ})/(1+R_{Z}^{2})^{3/2})-\rho\frac{1}{4}\frac{\partial}{\partial z}(R^{3}\ddot{R})=\rho R^{2}\ddot{Z} \qquad (6.2)$$

6.2 The Dimensionless Form of the Equation.

This equation has the physical dimensions of force per unit length and thus it becomes dimensionless when we divide by the surface tension, σ. Each of the physical quantities appearing in this equation can be expressed as a product of a corresponding nondimensional physical quantity and a dimensional constant formed of constant fundamental parameters of the problem; for mass, ρR_{0}^{3}; for length, R_{0}; and for time, $(\rho R_{0}^{3}/\sigma)^{1/2}$. By this procedure we can replace each variable of the equation with a dimensionless variable. For convenience, to avoid introducing new symbols, we retain the symbol of the dimensional quantity to represent the new dimensionless quantity. Thus, we make the following substitutions:

$$
\begin{array}{lll}
R\rightarrow R_0 R & Z\rightarrow R_0 Z & T\rightarrow \frac{\sigma}{R_0}T \\
R\rightarrow (\sigma/\rho\, R_0^2)R & Z\rightarrow(\sigma/\rho\, R_0^2)Z & F\rightarrow \sigma R_0 F \\
t\rightarrow(\rho R_0^3/\sigma)^{1/2}\, t & H_*\rightarrow(\rho\,\sigma^3 R_0^5)^{1/2}\, H_* & G_0\rightarrow \frac{\sigma}{R_0}G_0 \\
Z_0\rightarrow R_0 Z_0 & \alpha\rightarrow(\sigma/\rho\, R_0^3)^{1/2}\alpha & \text{etc.}
\end{array}
\qquad (6.3)
$$

where the symbols to the left of the arrows represent dimensional quantities which are represented in dimensionless form on the right by the same symbols multiplied by a dimensional constant. In terms of these dimensionless variables equation (6.2) takes on the following form:

$$\frac{\partial}{\partial z}(R^2 T)+\frac{\partial}{\partial z}(R(1+R_Z^2+RR_{ZZ})/(1+R_Z^2)^{3/2})-\frac{1}{4}\frac{\partial}{\partial z}(R^3\,\ddot{R})= R^2\,\ddot{Z} \qquad (6.4)$$

For ease in calculating it is convenient to express the spatial derivatives of this equation in terms of values at a reference configuration fixed in the material, that is, it is convenient to transform to a Lagrangian form. Accordingly, equation (6.4) becomes

$$\frac{\partial}{\partial z_0}(T/\lambda)+\frac{\partial}{\partial z_0}(R(1+R_Z^2+RR_{ZZ})/(1+R_Z^2)^{3/2})-\frac{1}{4}\frac{\partial}{\partial z_0}(R^3\,\ddot{R})= \ddot{Z}(Z_0,t) \qquad (6.5)$$

where the derivatives of the radius are to be calculated from the following equations:

$$R_Z=R^2R_{Z_0}, \qquad R_{ZZ}=2R^3R_{Z_0}^2+R^4R_{Z_0Z_0} \qquad (6.6)$$

For the Maxwell model we can replace the dimensionless stress of this equation, T, with the following expression:

$$T=\alpha\, G_0\int_{-\infty}^{t}[(\lambda^2(Z_0;t,\tau)-1/\lambda(Z_0;t,\tau))/\exp(\alpha(t-\tau))]\;d\tau \qquad (6.7)$$

where the rheological properties, modulus, G_0, and reciprocal relaxation time, α , are in the nondimensional forms given by equations (6.3).

Equations (6.5), (6.6) and (6.7) combine to give the Lagrangian form of the differential equation we are to solve. It is significant that in this nondimensional form the only remaining physical parameters of the equation are those introduced through the viscoelastic stresses.

6.3 Boundary Conditions and Initial Conditions.

In order to formulate a well posed mathematical problem we must add to this differential equation some suitable boundary conditions and initial conditions. The boundary conditions we shall use arise from the symmetries imposed by the spatial periodicity of the solution. We choose a length, L, to be the constant half period of the disturbance. The complete solution to the boundary value problem is then taken to be the solution within this half period reflected about the end points. These conditions are important in forming spatial derivatives at the ends of the half period.

If we assume that the filament at time zero is at rest in a cylindrical configuration of long standing, it will be in a state of equilibrium. Our analysis will predict no motion even though the equilibrium is unstable. It is necessary to presuppose a small initial perturbation as part of the initial conditions to start off the motion towards drop formation. This disturbance is taken to be a small sinusoidal displacement so that at time zero the stretch along the axis of the cylinder conforms to the following equation:

$$\lambda(Z_0,0)=1-e\pi\cos(\pi R_0 Z_0/L) \qquad (6.8)$$

We assume further that at time zero the local velocity is zero everywhere so that all previous forces do not influence the subsequent motion. As a result of these assumptions, the viscoelastic stress can be separated into two parts, a contribution due to deformations from the original equilibrium to the present configuration and a further contribution due to deformations from the initial perturbation and subsequent configurations. Equation (6.7), when broken up in this way, takes on the following form:

$$T = G_0 \exp(-\alpha t)\ (\lambda^2(Z_0,t) - 1/\lambda(Z_0,t)) + \alpha G_0 \exp(-\alpha t) \int_0^t [(\lambda^2(Z_0;t,\tau) - 1/\lambda(Z_0;t,\tau))]\ \exp(\alpha\tau)\, d\tau \qquad (6.9)$$

wherein the first term arises from the stretch calculated from the initial to the present configuration. It is equal to the stress relaxation we would have observed for a stress relaxation experiment starting with the present stretch from time zero and held to the present time. The second term is an integral over stretches from subsequent configurations up to the present. The stretch, $\lambda(Z_0,t)$, and the relative stretch, $\lambda(Z_0;t,\tau)$, are defined by equations (4.1) and (4.2), respectively.

Equations (6.5), (6.9) and initial conditions of equation (6.8) with the boundary conditions imposed by periodicity constitute the system we will attempt to solve by numerical methods.

7. SCALING AND CONDITIONING FOR NUMERICAL CALCULATION

A further change of variables is convenient (but not necessary) in conditioning the differential equation for solution by the method of finite differences. In imposing boundary conditions another length, the half period, has been introduced into the problem. The differential equation describing the motion of the filament is independent of this length and in no way implies a periodic solution. We are, in effect, selecting from among the pool of solutions to the differential equation by imposing this requirement. The boundary conditions appropriate to other problems might not introduce such an extra length and, in that case, the further change of variables described below might not be appropriate.

Because we will be calculating spatial derivatives using finite difference methods, it is useful to divide the half period into a fixed number of equal segments and to express distances along the axis of the filament in terms of their ratio with the half period. Then the measure of distance along the half period will range between zero and one, independent of the particular half period we use. Plots of solutions for different half periods can then be easily compared on the same scale. With this convenience in mind, we introduce the ratio of the initial radius of the filament to the half period, K. We redefine the lengths along the axis yet again in terms of the following dimensionless variables:

$$Z \to Z/K,\ Z_0 \to Z_0/K,\ K=R_0/L \qquad (7.1)$$

where, as before, the quantities on the left of the arrows are to be replaced by the quantities on the right throughout the equations and, in the results, Z and Z_0 are to be interpreted as ratios of axial distance to half period. Equation (6.5) then takes tne following form:

$$\frac{\partial}{\partial Z_0}(T/\lambda)+ + \frac{\partial}{\partial Z_0}(R(1+K^2R^4(3R_{Z_0}^2 +RR_{Z_0Z_0}))/(1+K^2R^4R_{Z_0}^2)^{3/2})- -\frac{1}{4}\frac{\partial}{\partial Z_0}(R^3\ddot{R})=\ddot{Z}(Z_0,t)/K^2 \qquad (7.2)$$

In terms of these new variables the initial conditions and the perturbation of equation (6.8) now take on the following simple forms:

$$Z(Z_0,0)=Z_0- e\ \sin(\pi Z_0),\ \lambda(Z_0,0)=1\ -e\pi\cos(\pi Z_0) \qquad (7.3)$$

The radial inertia term of equation (7.2) should be expressed in terms of spatial derivatives of axial displacement, velocity and acceleration. This can be done through the equation expressing the incompressibility of the medium, equation (5.1), which leads to the following:

$$\frac{\partial}{\partial Z_0}(R^3\ddot{R})= \frac{3}{2}R^8 Z_{Z_0Z_0}A_{Z_0} -\frac{R^6}{2}A_{Z_0Z_0} -3R^{10}Z_{Z_0Z_0}V_{Z_0}^2 +\frac{3}{2}R^8V_{Z_0}V_{Z_0Z_0} \qquad (7.4)$$

where $V=\dot{Z}$ and $A=\ddot{Z}$.

Equation (7.2) then takes the following form:

$$\frac{\partial}{\partial z_0}(T/\lambda)+ \\ +\frac{\partial}{\partial z_0}[R(1+K^2R^4(3R^2_{z_0}+RR_{z_0z_0}))/(1+K^2R^4R^2_{z_0})^{3/2}]+ \\ +\frac{3}{4}R^{10}Z_{z_0z_0}V^2_{z_0}-\frac{3}{8}R^8V_{z_0}V_{z_0z_0}= \\ =\frac{3}{8}R^8Z_{z_0z_0}A_{z_0}-\frac{R^6}{8}A_{z_0z_0}+A/K^2 \tag{7.5}$$

Equation (7.5) is in a form such that all terms in acceleration appear on the right hand side only, while the left hand side is calculable from the history of axial displacement (for calculating the viscoelastic force gradient) and the axial displacement and velocity and their derivatives. This form is suitable for numerical calculation using finite difference methods for evaluating derivatives and integrating by use of the trapazoid rule.

The half period L is divided up into N-1 equal intervals by distributing N nodes along the axis of the filament in its initial equilibrium configuration. The nodes are fixed in the material and the Ith node is characterized by $Z_0(I)$, its original equilibrium position. The quantities R, Z, V and A may be approximated, then, by N dimensional vectors R(I), Z(I), V(I) and A(I) whose components are the values at each node. The axial derivatives of these vectors are evaluated at each node by using the five point central difference equations of Appendix A. The stress gradient is calculated at each node by integrating in time using the trapezoid rule. In this way, equation (7.5) may be interpreted as a vector equation in the N dimensional space of nodes of the half period.

8. METHOD OF SOLVING THE INTEGRO-DIFFERENTIAL EQUATION

8.1 Integration of the Differential Equation.

The differential equation (7.5) is integrated in time by a method which might be described as using a "predictor/corrector" method to march in time. For the first step, the initial conditions give the displacement and velocity of each node at time zero. The left hand side of the equation can be evaluated from these quantities and their axial derivatives which are calculated by the methods of appendix I. The right hand side of equation (7.5) involves the unknown acceleration and its axial derivatives. Through the methods of appendix I this side may be expressed as an NxN matrix acting on the unknown acceleration vector. Because we use the five point central difference approximations for calculating the derivatives, the matrix is pentadiagonal, that is, the only nonzero elements of the matrix are in a band of five central diagonals. The set of N linear equations represented by the resulting vector equation can be solved easily with the technique explained in appendix II. This procedure can be used to calculate the acceleration at time zero.

With the acceleration at time zero we can estimate the initial motion. For a first estimate we assume that the acceleration remains constant and calculate for each node the corresponding displacement and velocity at the end of some small interval of time. With these quantities we can again use equation (7.5) in the same way as before to calculate a new estimate of acceleration at the end of the time interval. We then assume that the acceleration actually changed linearly in time from the initial value to the average of the two estimated values. We again calculate the corresponding displacement and velocity at each node and then the acceleration at the end of the small time interval. If this third estimated acceleration is not sufficiently close to the second estimate, we continue the process of averaging estimates and calculating new displacements, velocities and new estimated accelerations until two successive estimates of acceleration are sufficiently close to be considered equal. At this point we judge the calculation to be acceptable and the latest values of the variables are assigned to the vectors representing the physical quantities after one time step.

The next time step is calculated in the same way, except that for the initial trial we estimate that the acceleration varies linearly in time with a rate equal to the rate of the first time step. Subsequent trials assume a quadratic variation of acceleration fitted to the values at the two earlier times and the average of the two latest estimates for the value at the end of the time step.

The third and subsequent time steps are integrated assuming a quadratic variation of acceleration for the initial calculation and a cubic variation for calculations thereafter. In order to use this scheme the Fortran program keeps track of the four values of acceleration at each node; for two time steps into the past, the present acceleration and the current estimate for one step into the future. The appropriate extrapolations for acceleration, velocity and displacement in terms of these values are given in Appendix C.

8.2 Evaluating the Integral.

In calculating the gradient of the viscoelastic force we must perform an integration over the history of the deformation. For example, in the case of the Maxwell model the gradient of the force can be written as follows:

$$\begin{aligned}\frac{\partial}{\partial Z_0}(T/\lambda) &= (1+2/\lambda^3(Z_0,t))\lambda_{Z_0}(Z_0,t)\exp(-\alpha t) + \\ &+ \alpha\exp(-\alpha t)\int_0^t Q(Z_0;t,\tau)\ \exp(\alpha\tau)d\tau - \\ &- \alpha\exp(-\alpha t)\int_0^t Q(Z_0;\tau,t)\ \exp(\alpha\tau)d\tau \end{aligned} \qquad (8.1)$$

$$Q(Z_0;t,\tau) = [\lambda^{-2}(Z_0,\tau)+2\lambda(Z_0,\tau)/\lambda^3(Z_0,t)]\lambda_{Z_0}(Z_0,t)$$

The integrals of this equation are calculated by the Fortran program using the trapezoid rule. To be able to do so, the program stores the history of the displacement gradient and second derivative of displacement in the matrix EP. If no provision is made to discard old history, after a large number of time steps this information will require unacceptable amounts of storage. Fortunately, for any reasonable material, configurations very long in the past no longer contribute significantly to current stress. The program takes advantage of this circumstance by providing a means to discard data after an arbitrary number of time steps. This is accomplished by setting the value of the parameter M0. After the matrix EP has been completely filled the new data are inserted by replacing the oldest data. By using modular arithmetic for setting the indices of EP the data can be appropriately located.

9. HOW TO RUN FORTRAN PROGRAM DROPGEN

9.1 How to Set Parameters.

The program DROPGEN was developed on a Cyber 855 at the National Bureau of Standards. It is written for a Fortran 77 compiler and should run on most mainframe computers with suitable software. For any interested researcher the author will cooperate in transferring a copy by whatever means seems suitable.

DROPGEN is liberally annotated so that with a little patient study its construction and method of calculation should become clear to the reader. In preparing to run it, the parameters of a particular calculation are inserted directly into the beginning of the program at the places marked by comments. The number of nodes to be carried in the calculation is set by the quantity N0 and the size of the time steps is set by DELT. Experience and some trial and error are called for in setting these two parameters. Of course, the correct solution is independent of these quantities which have to do only with the numerical method. However, if too many nodes are carried or the time step is too small, the finite differences occuring in the calculations will suffer serious round-off error; if too few nodes or too large a time step, the calculation becomes too crude and results depend strongly on the choice of N0 and DELT. In most cases, N0=20 and DELT=0.1 are good guesses for a first trial. In any event, DELT must be smaller than the shortest relaxation time of importance. MEM is the length of the history of deformation carried in the calculation and should be longer than the longest significant viscoelastic relaxation time of the material. ITLIM limits the number of iterations allowed in calculating a time step. It serves to avoid the possibility of the calculation entering an infinite loop and does not affect a correct calculation in any way. Ordinarily, setting ITLIM=25 is convenient. The computation proceeds for a number of time steps determined by NTS. In some cases, however, the calculations lead to a sharp singularity in the surface of the jet followed by failure of the computation and termination before the requested number of time steps.

The mechanical parameters of the problem are set in a straightforward way. PTB is set to the amplitude of the sinusoidal perturbation and RRHP to the ratio of the radius of the original cylindrical jet to a half-period of the disturbance. The parameters of the Maxwell model are set by G0 and ALPHA. All sections depend on the BKZ model are marked off in DROPGEN by dotted lines so that it is convenient to modify the program to accomodate BKZ models with various kernal functions.

The parameter INTRVL controls the output of the DROPGEN by determining the number of time steps in the intervals between printed results.

9.2 How to Read the Printout.

A sample printout of DROPGEN appears in section 12. of this report. It begins with a self-explanatory listing of the values of the parameters for the calculation and a display of the radius, Z coordinate and viscoelastic stress at the time of the initial perturbation. Following this, the evolution of the jet is displayed by tablulations of the conditions at time intervals determined by the parameter INTRVL. Each tabulation is headed by a listing of the time lapse since perturbation and two indicators of the quality of the calculation. ITER indicates the maximum number of iterations needed in calculating the tme step. If ITER is equal to ITLIM then BADFIT indicates the number of nodes for which the accuracy criterion was not met even after ITLIM iterations. If BADFIT is not zero the calculation of that and subsequent time steps can not be trusted. The first three columns of the tabulation give the radius, Z coordinate and stretch, respectively at each node of the jet. The next three columns give (in order) the viscoelastic force gradient, surface tension force gradient and the gradient of the velocity-dependent terms of the force due to radial inertia. These quantities help one to understand the physics of the situation by indicating the relative importance of the various influences. The final two columns of the tabulation give the computed acceleration of each node and the difference between the computed and projected acceleration.

10. LIST OF SYMBOLS IN PROGRAM DROPGEN

AEST(I) A projection one time step into the future of the acceleration at a node based on a polynomial fit to the past accelerations. This projection is averaged with AX(I) to estimate AF(I).

AF(I) The acceleration of the Ith node at one time step into the future.

FPI,FPJ1,FPL Floating point equivalents of indices I,J1,L.

ALPHA The reciprocal of the dimensionless relaxation time of a Maxwell model. (see equation (6.3))

AO(I) The old acceleration of the Ith node at two time steps into the past.

AP(I) The present acceleration of the Ith node.

AR(I) The recent acceleration of the Ith node at one time step into the past.

AX(I) The acceleration of the Ith mode extrapolated one time step into the future.

BADFIT A count at each time-step of the nodes for which the iteration scheme does not converge sufficiently within ITMAX iterations.

BKZ(I) The gradient of the BKZ viscoelastic force at the Ith node. An exception occurs in the routine for printing out the initial conditions when this variable is temporarily used to store viscoelastic force.

C(I,J) A six by N0 matrix representing, in compact form, a finite difference operator corresponding to equation (7.5). C contains the coefficients of a pentadiagonal N0xN0 matrix and a 1xN0 vector of the inhomogeneous terms and it operates on the vector AX.

DELT The size of a time step.

DELTA(I) A coefficient vector used in subroutine VBAND.

DELZ The space between nodes as a fraction of the half-period of the perturbation.

DSPLP(I) The present displacement of the Ith node.

DSPLR(I) The displacement of the Ith node one time step into the past.

E(I,J) The Ith derivative of Z w.r.t. Z0 at the Jth node at the present time.

EP(I,J,K) The Ith derivative of Z w.r.t. Z0 at the Jth node at the Kth time step. The history of the deformation of the material is stored in EP.

EPSIL(I) A coefficient vector used in subroutine VBAND.

F1,F2 Auxiliary quantities used in calculating the radial acceleration term.

G(I) An auxiliary vector used in the subroutines.

GAMMA(I) A coefficient vector used in subroutine VBAND.

G0 The dimensionless, infinitesimal, instantaneous shear modulus of a Maxwell model. (see equation (6.3))

I,J,J1,K,L Integer indices for Fortran DO loops.

INTRVL The number of time steps elapsed between printing out the results of calculation.

ITER The number of the current iteration.

ITLIM The limit on the number of iterations allowed in calculating a time step at a node.

M1 The total number of time steps set in calculating a stress integral by trapezoid rule.

MEM The number of memory steps to be carried in evaluating the stress integral.

N0 The number of nodes.

NTS The number of time steps to be calculated.

PTB The amplitude of the initial perturbation.

Q A marker which keeps track of time intervals into the past during the evaluation of the visco-elastic stress integral.

R(I) The radius at the Ith mode.

RA An auxiliary quantity for calculating radial inertia.

RI(I) The sum of radial inertial terms which depend on gradients of the velocity

RRHP The ratio of radius of the equilibrium cylinder to the half-period of the disturbance.

RZ,RZZ,etc. The derivatives of radius w.r.t. spatial position.

S(I,J) The Ith derivative of the velocity of the Jth node w.r.t. Z0.

S1 Term to be summed in using trapezoid rule to evaluate integral.

SFT(I) The gradient of the force due to surface tension.

STRCH(I) The stretch at the Ith node.

SUM The partial sum during the evaluation of an integral.

T The present time.

T1,T2,T3,TR1 Temporary storage and for scratch pad usage.

TAU The time interval into the past at each step of evaluating the stress integral by trapezoid rule.

VP(I) The present velocity of the Ith node.

VR(I) The recent velocity of the Ith node one time step into the past.

W(I) An auxiliary vector used for scratch purposes.

XI(I) A coefficient vector used in subroutine VBAND.

Z0(I) The axial position of the Ith node in the equilibrium cylinder; a material label of the Ith node.

11. LISTING OF FORTRAN PROGRAM DROPGEN

```
                  PROGRAM DROPGEN

C  A FINITE DIFFERENCE PROGRAM TO CALCULATE THE FORMATION OF
C  DROPLETS IN A JET DUE TO THE COMBINED EFFECTS OF INERTIA,
C  SURFACE TENSION AND BKZ TYPE NONLINEAR VISCOELASTICITY.
C  THE JET IS MODELED AS AN INFINITE CYLINDER SUDDENLY PERTURBED
C  BY A SMALL SINUSOIDAL DISTURBANCE. A SECTION OF JET A HALF
C  WAVELENGTH LONG REPRESENTS THE EVOLUTION OF THE WHOLE JET.

C  SET PARAMETERS HERE

C    NO IS NUMBER OF NODES
C    NTS IS NUMBER OF TIME STEPS TO BE CALCULATED
C    MEM IS LENGTH OF MEMORY TO BE CARRIED (NUMBER OF TIME STEPS)
C    ITLIM IS LIMIT ON NUMBER OF ITERATIONS ALLOWED

      PARAMETER(NO=21,NTS=450,MEM=200,ITLIM=25)

      DIMENSION ZO(NO),DSPLR(NO),STRCH(NO),BKZ(NO),R(NO),W(NO)
      DIMENSION VR(NO),VP(NO),AP(NO),SFT(NO),E(4,NO),G(NO)
      DIMENSION AO(NO),AR(NO),EP(2,NO,MEM),AEST(NO),AF(NO)
      DIMENSION S(2,NO),C(6,NO),RI(NO),AX(NO),DSPLP(NO)
      DIMENSION GAMMA(NO-2),DELTA(NO-2),EPSIL(NO-2),XI(NO-2)

C  MECHANICAL PARAMETERS
C    DELT IS SIZE OF TIME STEP
C    PTB IS AMPLITUDE OF PERTURBATION
C    RRHP IS RATIO OF RADIUS TO HALFPERIOD
C    BEWARE OF INTEGER DIVISION! ALL ENTRIES WITH DECIMAL POINT.
      DELT=0.1
      PTB=0.01
      RRHP=1.0/20.0
C -------------------------------------------------------------
C  BKZ PARAMETERS

C    1/ALPHA IS HALF-LIFE OF STRESS RELAXATION
C    GO IS YOUNG'S MODULUS
C    BEWARE OF INTEGER DIVISION! ALL ENTRIES WITH DECIMAL POINT.
      ALPHA=0.5
      GO=0.2
C -------------------------------------------------------------

C  INTRVL IS NUMBER OF TIME STEPS BETWEEN PRINT-OUTS
      INTRVL = 15

C PRINT OUT PARAMETERS
      OPEN(6,FILE='TAPE6')
      WRITE(6,73)
      WRITE(6,83) NO,NTS,MEM
83    FORMAT(3X,I4,6X,I4,11X,I4/)
73    FORMAT(5X,'NODES',5X,'NO. TSTEPS',4X,'MEMORY')
```

```
      WRITE(6 ,250)
250   FORMAT(5X,'GO',7X,'ALPHA',5X,'RAD/HALFP',6X,'DELT',5X,'PERTURB')
      WRITE(6 ,260) GO,ALPHA,RRHP,DELT,PTB
260   FORMAT(1X,F8.4,2X,F9.3,3X,F8.5,7X,F6.4,3X,F6.4,3X,F5.2/)
C HEADING FOR INITIAL CONDITIONS PRINT OUT
      WRITE(6,*) 'STARTING CONDITIONS'
      WRITE(6 ,200)
200   FORMAT(3X,'RADIUS',5X,'Z COORD',4X,'STRESS')
C  PUT IN INITIAL DISTURBANCE
      DO 20 I=1,NO
      FPI=I
      AR(I)=0.0
      AO(I)=0.0
      VR(I)=0.0
      DELZ=1.0/(NO-1)
      ZO(I)=DELZ*(FPI-1.0)
      PI=3.1415926535898
      DSPLP(I)=-PTB*SIN(PI*ZO(I))
      DSPLR(I)=DSPLP(I)
      VP(I)=0.0
      STRCH(I)=1.0-PI*PTB*COS(PI*ZO(I))
C  ---------------------------------------
C     CALCULATE INITIAL STRESS
      R(I)=1.0/SQRT(STRCH(I))
      BKZ(I)=STRCH(I)**2-1.0/STRCH(I)
C  ---------------------------------------
C  PRINT OUT INITIAL CONDITIONS
      WRITE(6,210) R(I),ZO(I)+DSPLR(I),BKZ(I)
210   FORMAT(2X,F8.6,2X,F8.5,2X,F10.6)
20    CONTINUE

C                    CALCULATE INITIAL ACCELERATION

C CALCULATE SURFACE TENSION TERM
      CALL DERIV(DSPLP,NO,E)
      DO 30 I=1,NO
      EP(1,I,1)=E(1,I)
      EP(2,I,1)=E(2,I)
      RZ=-R(I)*E(2,I)/(2.0*E(1,I)**2)
      RZZ=R(I)*(5.0*E(2,I)**2-2.0*E(1,I)*E(3,I))
      RZZ=RZZ/(4.0*E(1,I)**4)
      RZZZ=34.0*E(1,I)*E(2,I)*E(3,I)-45.0*E(2,I)**3
      RZZZ=R(I)*(RZZZ-4.0*E(1,I)**2*E(4,I))/(8.0*E(1,I)**6)

      T3=SQRT(1+RRHP**2*RZ**2)
      T2=RZ/T3
      T2=T2+RRHP**2*R(I)*(RZ*RZZ+R(I)*RZZZ)/T3**3
      T2=T2-3.0*RRHP**4*R(I)**2*RZ*RZZ**2/T3**5
      T2=E(1,I)*T2

C   -------------------------------------
C CALCULATE VISCOELASTIC FORCE GRADIENT
      T1=(1.0+2.0/E(1,I)**3)*E(2,I)*GO*RRHP**2
```

```
C    ----------------------------------------

C CALCULATE INITIAL ACCELERATION
      AR(I)=0.0
      AP(I)=T1+T2*RRHP**2
30    CONTINUE

C                           DO A TIME STEP
      DO 50 J1=1,NTS

C  SET LENGTH OF FINITE MEMORY
      J=MOD(J1-1,MEM)+1

C  CALCULATE TIME
      FPJ1=J1
      T=DELT*FPJ1

C CALCULATE STRETCH

      ITER=-1
C ENTRY FOR ITERATION
816   BADFIT=0
      ITER=ITER+1
      INDEX=ITER-1
      CALL DERIV(DSPLP,NO,E)

      DO 60 I=1,NO
C RECORD HISTORY OF STRETCH AND GRADIENT (IN MATRIX EP)
      EP(1,I,J)=E(1,I)
      EP(2,I,J)=E(2,I)

C CALCULATE STRETCH AND RADIUS
      STRCH(I)=E(1,I)
      R(I)=1.0/SQRT(E(1,I))

C    -----------------------------------------------------
C                  CALCULATE BKZ FORCE GRADIENT

      TR1=EXP(-ALPHA*T)*(1.0+2.0/E(1,I)**3)*E(2,I)
C SET NUMBER OF STEPS OF MEMORY TO BE INCLUDED
      IF (J1.LT.MEM) THEN
      M1=J-1
      ELSE
      M1=MEM-1
      ENDIF
C                  INTEGRATION USING TRAPEZOIDAL RULE
      SUM=0.0
      DO 70 L=1,M1
C SET Q TO EXTRACT DATA IN PAST FROM EP
      Q=MOD(MEM+J-L-1,MEM)+1

      EL1=EP(1,I,Q)
      EL2 =EP(2,I,Q)
      FPL=L
      TAU=DELT*FPL
      S1=(1.0/EL1**2+2.0*EL1/E(1,I)**3)*E(2,I)
      S1=S1-(1.0/E(1,I)**2+2.0*E(1,I)/EL1**3)*EL2
      S1=S1*EXP(-ALPHA*TAU)

      IF(L.NE.1) GO TO 80
      SUM=SUM+S1
      GO TO 70
80    IF(L.NE.M1) GO TO 90
      SUM=SUM+S1
      GO TO 70
90    SUM=SUM+2.0*S1
70    CONTINUE
      BKZ(I)=TR1+SUM*DELT/2.0
60    CONTINUE
C ------------------------------------------------

C                    CALCULATE RADIAL INERTIA TERM

      CALL SLOPES(VP,NO,S)

      DO 197 K=1,NO
      C(1,K)=0.0
      C(2,K)=0.0
      C(3,K)=0.0
      C(4,K)=0.0
      C(5,K)=0.0
      C(6,K)=0.0
      F1=3.0*E(2,K)*RRHP**2/(8.0*E(1,K)**4*DELZ)
      F2=-RRHP**2/(8.0*E(1,K)**3*DELZ**2)
      IF (K.EQ.1) THEN
      C(3,1)=1.0-F1*7.0/6.0
      C(4,1)=4.0*F1/3.0
      C(5,1)=-F1/6.0
      ELSE IF (K.EQ.2) THEN
      C(2,2)=7.0*F2/6.0-F1/2.0
      C(3,2)=1.0-F1/12.0-29.0*F2/12.0
      C(4,2)=2.0*F1/3.0+4.0*F2/3.0
      C(5,2)=-(F1+F2)/12.0
      ELSE IF (K.EQ.NO) THEN
      C(1,NO)=F1/6.0
      C(2,NO)=-4.0*F1/3.0
      C(3,NO)=1.0+7.0*F1/6.0
      ELSE IF (K.EQ.NO-1) THEN
      C(1,NO-1)=(F1-F2)/12.0
      C(2,NO-1)=4.0*F2/3.0-2.0*F1/3.0
      C(3,NO-1)=1.0+(F1-29.0*F2)/12.0
      C(4,NO-1)=F1/2.0+7.0*F2/6.0
      ELSE
      C(1,K)=(F1-F2)/12.0
      C(2,K)=(4.0*F2-2.0*F1)/3.0
      C(3,K)=1.0-5.0*F2/2.0
      C(4,K)=(4.0*F2+2.0*F1)/3.0
      C(5,K)=-(F1+F2)/12.0
      ENDIF
C VELOCITY DEPENDENT RADIAL INERIA TERMS
      RA=-6.0*S(1,K)**2*E(2,K)/E(1,K)**5
      RA=RA+3.0*S(1,K)*S(2,K)/E(1,K)**4

      RI(K)=RA*RRHP**2/8.0


197   CONTINUE

C                  CALCULATE SURFACE TENSION FORCE GRADIENT
      DO 110 I=1,NO
      RZ=-R(I)*E(2,I)/(2.0*E(1,I)**2)
      RZZ=R(I)*(5.0*E(2,I)**2-2.0*E(1,I)*E(3,I))
      RZZ=RZZ/(4.0*E(1,I)**4)
      RZZZ=34.0*E(1,I)*E(2,I)*E(3,I)-45.0*E(2,I)**3
      RZZZ=R(I)*(RZZZ-4.0*E(1,I)**2*E(4,I))/(8.0*E(1,I)**6)

      T3=SQRT(1+RRHP**2*RZ**2)
      T2=RZ/T3
      T2=T2+RRHP**2*R(I)*(RZ*RZZ+R(I)*RZZZ)/T3**3
      T2=T2-3.0*RRHP**4*R(I)**2*RZ*RZZ**2/T3**5
      T2=E(1,I)*T2

C CALCULATE SIXTH COLUMN OF CMAT
      T1=BKZ(I)*GO*RRHP**2
      SFT(I)=T2*RRHP**2
      C(6,I)=T1+T2*RRHP**2-RI(I)
110   CONTINUE

C INVERT CMAT TO GET NEW ACCELERATION
      CALL VBAND(C,NO,AX,GAMMA,DELTA,EPSIL,XI)

C                   RECALCULATE VELOCITY AND DISPLACEMENT

C              FOR EACH TIME STEP, THE FIRST ESTIMATE (ITER=0) OF
C              ACCELERATION, AEST, IS AN UNAVERAGED PROJECTION.

      DO 999 I=1,NO

C FIRST TIME STEP
            IF (J1.EQ.1) THEN

      IF (ITER.EQ.0) THEN
      AEST(I)=AP(I)
        ELSE
      AEST(I)=AF(I)
        ENDIF

      AF(I)=(AEST(I)+AX(I))/2.0
      VP(I)=VR(I)+AF(I)*DELT
      DSPLP(I)=DSPLR(I)+DELT*VR(I)+DELT**2*AF(I)/2.0

C SECOND TIME STEP
            ELSE IF (J1.EQ.2) THEN

        IF (ITER.EQ.0) THEN
      AEST(I)=2.0*AP(I)-AR(I)
        ELSE
      AEST(I)=AF(I)
        ENDIF

      AF(I)=(AEST(I)+AX(I))/2.0
      VP(I)=VR(I)+(13.0*AP(I)-2.0*AF(I)-5.0*AR(I))*DELT/6.0
      DSPLP(I)=26.0*AP(I)-9.0*AR(I)-5.0*AF(I)
      DSPLP(I)=DSPLR(I)+DELT*VR(I)+DELT**2*DSPLP(I)/24.0

C THIRD AND SUBSEQUENT TIME STEPS
            ELSE

        IF (ITER.EQ.0) THEN
      AEST(I)=3.0*(AP(I)-AR(I))+AO(I)
        ELSE
      AEST(I)=AF(I)
        ENDIF

      AF(I)=(AEST(I)+AX(I))/2.0
      VP(I)=9.0*AF(I)+AO(I)-5.0*AR(I)+19.0*AP(I)
      VP(I)=VR(I)+DELT*VP(I)/24.0
      DSPLP(I)=38.0*AF(I)+7.0*AO(I)-36.0*AR(I)+171.0*AP(I)
      DSPLP(I)=DSPLR(I)+DELT*VR(I)+DELT**2*DSPLP(I)/360.0

            ENDIF

C              CHECK ACCURACY OF PREDICTIONS
C TO AVOID DIVISION BY ZERO
      IF ((I-1)*(NO-I).EQ.0) GO TO 999

C CRITERION FOR GOOD FIT
      IF (ABS(AX(I)/AEST(I)-1.0).LT.0.00001) GO TO 999

C COUNT THE NODES WITH BAD PREDICTIONS
      BADFIT = BADFIT + 1
999   CONTINUE
C LIMIT NUMBER OF ITERATIONS TO AVOID INFINITE LOOP
      IF (ITER.GT.ITLIM) GO TO 500
C IF PREDICTED ACCELERATIONS ARE NOT ACCURATE AT ANY NODES, ITERATE
      IF (BADFIT.GT.0) GO TO 816
500   CONTINUE

C                   RESET KINEMATIC QUANTITIES
C RESET ACCELERATIONS
C AO-OLDEST, AR-RECENT, AP-PRESENT, AF-FUTURE
      DO 111 I=1,NO
      AO(I)=AR(I)
      AR(I)=AP(I)
      AP(I)=AF(I)
C RESET V AND H
C DISPLACEMENTS, DSPLP-PRESENT, DSPLR-RECENT
C VELOCITIES, VP-PRESENT, VR-RECENT
      W(I)=DSPLR(I)
      VR(I)=VP(I)
      DSPLR(I)=DSPLP(I)
111   CONTINUE

C  PRINT OUT RESULTS

      IF(MOD(J1,INTRVL).NE.0) GO TO 50
      IBAD=BADFIT
      WRITE(6,271) T,ITER,IBAD
271   FORMAT(/3X,'TIME=',F9.5,3X,'ITER=',I4,3X,'BADFIT=',I5)
      WRITE(6,40)
      SFT(1)=0.0
      SFT(NO)=0.0
      DO 293 I=1,NO
      WRITE(6,230)R(I),ZO(I)+W(I),STRCH(I),BKZ(I),SFT(I)/RRHP**2,-RI(I)
     +,AEST(I),AX(I)-AEST(I)
293   CONTINUE
50    CONTINUE
      GO TO 400
40    FORMAT(3X,'RADIUS',4X,'Z COORD',5X,
     +'LAMBDA',2X,'FORCE GRAD',3X,'SRFTEN GRD',3X,'RAD INRT',
     +5X,'AEST',5X,'A-AEST')
230   FORMAT(2X,F8.5,2X,F8.5,3X,F7.5,3X,F10.5,2X,
     +F10.5,2X,F8.5,2X,F9.5,2X,E15.6)
400   CONTINUE
      STOP
      END


C                    SUBPROGRAM DERIV
C  CALCULATES THE DERIVATIVES OF Z WRT ZO USING
C  FIVE POINT CENTRAL DIFFERENCES.

      SUBROUTINE DERIV(G,NO,E)
      DIMENSION G(*),E(4,*)
      DELZ=1.0/(NO-1)

      DO 7 I=1,NO
      IF (I.EQ.1) THEN
      E(1,I)=(8.0*G(2)-7.0*G(1)-G(3))/(6.0*DELZ)+1
      E(2,I)=0
      E(3,I)=(G(3)+G(1)-2.0*G(2))/DELZ**3
      E(4,I)=0

      ELSE IF (I.EQ.2) THEN
      E(1,I)=(8.0*G(3)-6.0*G(1)-G(4)-G(2))/(12.0*DELZ)+1
      E(2,I)=(16.0*G(3)+14.0*G(1)-G(4)-29.0*G(2))/(12.0*DELZ**2)
      E(3,I)=(G(4)+G(2)-2.0*G(3))/(2.0*DELZ**3)
      E(4,I)=(G(4)-2.0*G(1)+5.0*G(2)-4.0*G(3))/DELZ**4

      ELSE IF (I.EQ.NO) THEN
      E(1,I)=(7.0*G(NO)-8.0*G(NO-1)+G(NO-2))/(6.0*DELZ)+1
      E(2,I)=0
      E(3,I)=(2.0*G(NO-1)-G(NO)-G(NO-2))/DELZ**3
      E(4,I)=0

      ELSE IF (I.EQ.NO-1) THEN
      E(1,I)=(6.0*G(NO)-8.0*G(NO-2)+G(NO-1)+G(NO-3))/(12.0*DELZ)+1
      E(2,I)=(14.0*G(NO)+16.0*G(NO-2)-G(NO-3)-29.0*G(NO-1))/12.0
      E(2,I)=E(2,I)/(DELZ*DELZ)
      E(3,I)=(2.0*G(NO-2)-G(NO-1)-G(NO-3))/(2.0*DELZ**3)
      E(4,I)=(5.0*G(NO-1)-2.0*G(NO)+G(NO-3)-4.0*G(NO-2))/DELZ**4

      ELSE
      E(1,I)=(8.0*G(I+1)-8.0*G(I-1)-G(I+2)+G(I-2))/(12.0*DELZ)+1
      E(2,I)=(16.0*G(I+1)+16.0*G(I-1)-G(I+2)-G(I-2)-30.0*G(I))/12.0
      E(2,I)=E(2,I)/(DELZ*DELZ)
      E(3,I)=(G(I+2)-G(I-2)-2.0*(G(I+1)-G(I-1)))/(2.0*DELZ**3)
      E(4,I)=(G(I+2)+G(I-2)-4.0*(G(I+1)+G(I-1))+6.0*G(I))/DELZ**4

      ENDIF
7     CONTINUE

      RETURN
      END


C                  SUBPROGRAM SLOPES
C  CALCULATES THE FIRST AND SECOND DERIVATIVES
C  OF V WRT ZO USING FIVE POINT CENTRAL DIFF.

      SUBROUTINE SLOPES(G,NO,E)
      DIMENSION G(*),E(2,*)
      DELZ=1.0/(NO-1)

      DO 7 I=1,NO
      IF (I.EQ.1) THEN
      E(1,I)=(8.0*G(2)-7.0*G(1)-G(3))/(6.0*DELZ)
      E(2,I)=0

      ELSE IF (I.EQ.2) THEN
      E(1,I)=(8.0*G(3)-6.0*G(1)-G(4)-G(2))/(12.0*DELZ)
      E(2,I)=(16.0*G(3)+14.0*G(1)-G(4)-29.0*G(2))/(12.0*DELZ**2)

      ELSE IF (I.EQ.NO) THEN
      E(1,I)=(7.0*G(NO)-8.0*G(NO-1)+G(NO-2))/(6.0*DELZ)
      E(2,I)=0

      ELSE IF (I.EQ.NO-1) THEN
      E(1,I)=(6.0*G(NO)-8.0*G(NO-2)+G(NO-1)+G(NO-3))/(12.0*DELZ)
      E(2,I)=(14.0*G(NO)+16.0*G(NO-2)-G(NO-3)-29.0*G(NO-1))/12.0
      E(2,I)=E(2,I)/(DELZ*DELZ)

      ELSE
      E(1,I)=(8.0*G(I+1)-8.0*G(I-1)-G(I+2)+G(I-2))/(12.0*DELZ)
      E(2,I)=(16.0*G(I+1)+16.0*G(I-1)-G(I+2)-G(I-2)-30.0*G(I))/12.0
      E(2,I)=E(2,I)/(DELZ*DELZ)

      ENDIF
7     CONTINUE

      RETURN
      END


C                        SUBPROGRAM VBAND
C SOLVES A SYSTEM OF INHOMOGENEOUS LINEAR EQUATIONS WHICH
C ARE GIVEN IN TERMS OF A FIVE-DIAGONAL NOXNO MATRIX OF
C COEFFICIENTS. THE MATRIX C(6,NO) CONTAINS THE COEFFICIENTS
C OF THE SYSTEM AUGMENTED BY A VECTOR OF INHOMOGENEOUS TERMS.

      SUBROUTINE VBAND(C,NO,A,G,D,E,XI)

      DIMENSION C(6,*),A(*)
      DIMENSION G(*),D(*),E(*),XI(*)

C CALCULATE XI,GAMMA,DELTA,EPSILON

      XI(1)=C(3,1)
      G(1)=C(6,1)/XI(1)
      D(1)=-C(4,1)/XI(1)
      E(1)=-C(5,1)/XI(1)

      XI(2)=C(3,1)*C(3,2)-C(2,2)*C(4,1)
      G(2)=(C(3,1)*C(6,2)-C(2,2)*C(6,1))/XI(2)
      D(2)=(C(2,2)*C(5,1)-C(3,1)*C(4,2))/XI(2)
      E(2)=-C(3,1)*C(5,2)/XI(2)

      DO 73 I=3,NO-2
      XI(I)=C(1,I)*(D(I-1)*D(I-2)+E(I-2))
      XI(I)=XI(I)+C(2,I)*D(I-1)+C(3,I)
      G(I)=C(6,I)-C(1,I)*(G(I-2)+D(I-2)*G(I-1))
      G(I)=(G(I)-C(2,I)*G(I-1))/XI(I)
      D(I)=-(C(1,I)*D(I-2)*E(I-1)+C(2,I)*E(I-1))
      D(I)=(D(I)-C(4,I))/XI(I)
      E(I)=-C(5,I)/XI(I)
73    CONTINUE

C CALCULATE A(NO) AND A(NO-1)

      P1=C(2,NO)+C(1,NO)*D(NO-2)
      Q1=C(3,NO)+C(1,NO)*E(NO-2)
      R1=C(6,NO)-C(1,NO)*G(NO-2)

      P2=C(3,NO-1)+C(2,NO-1)*D(NO-2)
      P2=P2+C(1,NO-1)*(E(NO-3)+D(NO-2)*D(NO-3))
      Q2=C(4,NO-1)+C(2,NO-1)*E(NO-2)
      Q2=Q2+C(1,NO-1)*D(NO-3)*E(NO-2)
      R2=C(6,NO-1)-C(2,NO-1)*G(NO-2)
      R2=R2-C(1,NO-1)*(G(NO-3)+D(NO-3)*G(NO-2))

      DENOM=(P1*Q2-Q1*P2)
      A(NO)=(P1*R2-P2*R1)/DENOM
      A(NO-1)=(R1*Q2-Q1*R2)/DENOM

C CALCULATE A(I) FOR I LESS THAN NO-1

      DO 75 I=NO-2,1,-1
      A(I)=G(I)+D(I)*A(I+1)+E(I)*A(I+2)
75    CONTINUE

      RETURN
      END
```

12.   A SAMPLE PRINTOUT OF PROGRAM DROPGEN

```
   NODES      NO. TSTEPS     MEMORY
   21         150               10

   G0         ALPHA       RAD/HALFP        DELT      PERTURB
  .2000        .500        .05000          .3000     .0100
STARTING CONDITIONS
  RADIUS      Z COORD       STRESS
 1.016088     .00000     -.094280
 1.015885     .04844     -.093118
 1.015282     .09691     -.089662
 1.014297     .14546     -.083998
 1.012955     .19412     -.076265
 1.011296     .24293     -.066654
 1.009363     .29191     -.055404
 1.007208     .34109     -.042791
 1.004890     .39049     -.029125
 1.002466     .44012     -.014744
 1.000000     .49000      .000000
  .997552     .54012      .014744
  .995181     .59049      .029125
  .992944     .64109      .042790
  .990893     .69191      .055404
  .989074     .74293      .066654
  .987529     .79412      .076264
  .986291     .84546      .083997
  .985387     .89691      .089661
  .984837     .94844      .093116
  .984653    1.00000      .094278

TIME=  6.00000   ITER=  10   BADFIT=    0
RADIUS      Z COORD       LAMBDA  FORCE GRAD     SRFTEN GRD    RAD INRT        AEST        A-AEST
1.01750     .00000      .96590      .00000         .00000     .00000       .00000      .000000E+00
1.01728     .04832      .96632      .00540       -.00847     .00000      -.00002      .151845E-09
1.01662     .09668      .96756      .01063       -.01671     .00000      -.00004      .297608E-09
1.01555     .14513      .96960      .01553       -.02451     .00000      -.00005      .431535E-09
1.01410     .19369      .97239      .01996       -.03166     .00000      -.00007      .548515E-09
1.01230     .24241      .97585      .02379       -.03796     .00000      -.00008      .644386E-09
1.01020     .29131      .97992      .02692       -.04327     .00000      -.00009      .716194E-09
1.00785     .34043      .98448      .02925       -.04744     .00000      -.00010      .762368E-09
1.00533     .38978      .98942      .03075       -.05038     .00000      -.00011      .782802E-09
1.00269     .43939      .99463      .03140       -.05203     .00000      -.00011      .778788E-09
1.00001     .48926      .99998      .03123       -.05237     .00000      -.00011      .752813E-09
 .99735     .53939     1.00532      .03027       -.05141     .00000      -.00011      .708231E-09
 .99477     .58978     1.01054      .02859       -.04920     .00000      -.00011      .648868E-09
 .99234     .64042     1.01549      .02629       -.04581     .00000      -.00010      .578609E-09
 .99011     .69130     1.02007      .02344       -.04136     .00000      -.00009      .501034E-09
 .98814     .74240     1.02414      .02016       -.03596     .00000      -.00008      .419153E-09
 .98647     .79368     1.02763      .01652       -.02976     .00000      -.00007      .335247E-09
 .98513     .84512     1.03043      .01261       -.02290     .00000      -.00005      .250837E-09
 .98415     .89668     1.03248      .00851       -.01554     .00000      -.00003      .166735E-09
 .98355     .94832     1.03373      .00428       -.00786     .00000      -.00002      .831794E-10
 .98335    1.00000     1.03415      .00000         .00000     .00000       .00000      .000000E+00

TIME= 12.00000   ITER=  11   BADFIT=    0
RADIUS      Z COORD       LAMBDA  FORCE GRAD     SRFTEN GRD    RAD INRT        AEST        A-AEST
1.02579     .00000      .95034      .00000         .00000     .00000       .00000      .000000E+00
1.02546     .04759      .95096      .01026       -.01272     .00000      -.00003      .245825E-09
1.02447     .09523      .95279      .02012       -.02507     .00000      -.00005      .479344E-09
1.02286     .14300      .95580      .02919       -.03671     .00000      -.00008      .689181E-09
1.02067     .19093      .95991      .03715       -.04731     .00000      -.00010      .865748E-09
1.01796     .23910      .96502      .04371       -.05656     .00000      -.00012      .100196E-08
1.01483     .28753      .97099      .04868       -.06422     .00000      -.00014      .109371E-08
1.01135     .33628      .97769      .05193       -.07009     .00000      -.00015      .114014E-08
1.00762     .38536      .98493      .05344       -.07403     .00000      -.00016      .114345E-08
1.00375     .43481      .99254      .05330       -.07600     .00000      -.00016      .110851E-08
 .99984     .48463     1.00031      .05165       -.07598     .00000      -.00016      .104215E-08
 .99600     .53483     1.00805      .04871       -.07405     .00000      -.00016      .952214E-09
 .99231     .58541     1.01557      .04473       -.07034     .00000      -.00015      .846686E-09
 .98885     .63635     1.02267      .03997       -.06502     .00000      -.00014      .732825E-09
 .98572     .68762     1.02919      .03467       -.05828     .00000      -.00013      .616568E-09
```

```
  .98296     .73919   1.03498          .02905      -.05033     .00000     -.00011        .502198E-09
  .98063     .79102   1.03989          .02326      -.04140     .00000     -.00009        .392282E-09
  .97878     .84307   1.04383          .01742      -.03171     .00000     -.00007        .287820E-09
  .97743     .89528   1.04671          .01159      -.02145     .00000     -.00005        .188536E-09
  .97662     .94761   1.04846          .00578      -.01081     .00000     -.00002        .932062E-10
  .97635    1.00000   1.04904          .00000       .00000     .00000      .00000        .000000E+00

TIME= 18.00000    ITER=  12    BADFIT=    0
RADIUS     Z COORD      LAMBDA  FORCE GRAD      SRFTEN GRD    RAD INRT       AEST        A-AEST
1.04545     .00000     .91495          .00000       .00000     .00000      .00000        .000000E+00
1.04483     .04589     .91602          .02443      -.02312     .00000     -.00005        .326180E-09
1.04301     .09187     .91923          .04770      -.04551     .00000     -.00009        .631048E-09
1.04003     .13807     .92449          .06875      -.06647     .00000     -.00013        .895335E-09
1.03602     .18457     .93168          .08661      -.08534     .00000     -.00017        .110367E-08
1.03109     .23146     .94060          .10055      -.10151     .00000     -.00020        .124606E-08
1.02542     .27882     .95104          .11007      -.11449     .00000     -.00023        .131874E-08
1.01919     .32672     .96269          .11497      -.12390     .00000     -.00025        .132437E-08
1.01260     .37520     .97526          .11539      -.12954     .00000     -.00027        .127128E-08
1.00586     .42432     .98838          .11176      -.13136     .00000     -.00027        .117204E-08
  .99915     .47407   1.00169          .10475      -.12951     .00000     -.00027        .104148E-08
  .99266     .52447   1.01484          .09520      -.12430     .00000     -.00026        .894573E-09
  .98654     .57550   1.02746          .08401      -.11616     .00000     -.00025        .744530E-09
  .98093     .62712   1.03927          .07202      -.10559     .00000     -.00023        .601545E-09
  .97591     .67929   1.04998          .05992      -.09311     .00000     -.00020        .472222E-09
  .97158     .73195   1.05937          .04823      -.07919     .00000     -.00017        .359689E-09
  .96798     .78504   1.06726          .03724      -.06427     .00000     -.00014        .264195E-09
  .96515     .83847   1.07352          .02704      -.04867     .00000     -.00011        .183908E-09
  .96312     .89216   1.07806          .01757      -.03264     .00000     -.00007        .115698E-09
  .96189     .94604   1.08081          .00864      -.01638     .00000     -.00004        .557595E-10
  .96148    1.00000   1.08173          .00000       .00000     .00000      .00000        .000000E+00

TIME= 24.00000    ITER=  13    BADFIT=    0
RADIUS     Z COORD      LAMBDA  FORCE GRAD      SRFTEN GRD    RAD INRT       AEST        A-AEST
1.08743     .00000     .84567          .00000       .00000     .00000      .00000        .000000E+00
1.08616     .04255     .84765          .06090      -.04574     .00000     -.00008        .492466E-09
1.08239     .08528     .85356          .11845      -.09007     .00000     -.00016        .943276E-09
1.07627     .12840     .86329          .16943      -.13152     .00000     -.00024        .131490E-08
1.06805     .17208     .87664          .21093      -.16861     .00000     -.00031        .157812E-08
1.05805     .21649     .89329          .24060      -.19987     .00000     -.00038        .171593E-08
1.04667     .26177     .91281          .25694      -.22393     .00000     -.00043        .172644E-08
1.03437     .30806     .93465          .25956      -.23974     .00000     -.00047        .162363E-08
1.02161     .35545     .95813          .24937      -.24673     .00000     -.00049        .143494E-08
1.00887     .40399     .98248          .22859      -.24497     .00000     -.00050        .119567E-08
  .99657     .45373   1.00690          .20035      -.23525     .00000     -.00049        .941543E-09
  .98505     .50463   1.03059          .16822      -.21892     .00000     -.00046        .701953E-09
  .97457     .55665   1.05287          .13559      -.19771     .00000     -.00043        .496057E-09
  .96531     .60970   1.07317          .10512      -.17339     .00000     -.00038        .332328E-09
  .95734     .66369   1.09111          .07855      -.14754     .00000     -.00033        .210764E-09
  .95070     .71850   1.10641          .05662      -.12136     .00000     -.00028        .126099E-09
  .94535     .77398   1.11896          .03925      -.09561     .00000     -.00022        .707442E-10
  .94127     .83001   1.12869          .02586      -.07065     .00000     -.00016        .368461E-10
  .93839     .88645   1.13562          .01554      -.04654     .00000     -.00011        .174123E-10
  .93668     .94315   1.13977          .00726      -.02309     .00000     -.00005        .668540E-11
  .93612    1.00000   1.14115          .00000       .00000     .00000      .00000        .000000E+00

TIME= 30.00000    ITER=  14    BADFIT=    0
RADIUS     Z COORD      LAMBDA  FORCE GRAD      SRFTEN GRD    RAD INRT       AEST        A-AEST
1.18253     .00000     .71512          .00000       .00000     .00000      .00000        .000000E+00
1.17970     .03624     .71855          .16431      -.08939     .00000     -.00014        .904549E-09
1.17131     .07282     .72888          .32174      -.17795     .00000     -.00028        .173702E-08
1.15765     .11007     .74618          .46444      -.26427     .00001     -.00042        .242348E-08
1.13923     .14833     .77050          .58297      -.34572     .00001     -.00056        .289024E-08
1.11681     .18794     .80176          .66644      -.41800     .00001     -.00070        .307412E-08
1.09142     .22923     .83950          .70409      -.47512     .00000     -.00082        .294327E-08
1.06433     .27248     .88276          .68874      -.51028     .00000     -.00092        .252397E-08
1.03700     .31792     .92992          .62146      -.51803     .00000     -.00097        .191504E-08
1.01082     .36568     .97871          .51448      -.49709     .00000     -.00098        .126465E-08
  .98696     .41578   1.02660          .38889      -.45186     .00000     -.00093        .708748E-09
  .96618     .46812   1.07123          .26708      -.39121     .00000     -.00084        .315406E-09
  .94877     .52250   1.11090          .16498      -.32501     .00000     -.00073        .797861E-10
  .93466     .57868   1.14471          .08914      -.26108     .00000     -.00061       -.403415E-10
  .92352     .63637   1.17248          .03865      -.20394     .00000     -.00049       -.894406E-10
  .91495     .69529  1.19456       .00879      -.15514     .00000     -.00038   -.993939E-10
  .90851     .75516  1.21154      -.00611      -.11434     .00000     -.00029   -.893934E-10
  .90387     .81577  1.22403      -.01109      -.08014     .00000     -.00021   -.701414E-10
  .90073     .87690  1.23255      -.01004      -.05083     .00000     -.00013   -.473307E-10
  .89893     .93837  1.23752      -.00572      -.02465     .00000     -.00006   -.236567E-10
  .89834    1.00000  1.23915       .00000       .00000     .00000      .00000    .000000E+00

TIME= 36.00000   ITER=  16   BADFIT=     0
RADIUS     Z COORD   LAMBDA   FORCE GRAD   SRFTEN GRD   RAD INRT        AEST   A-AEST
1.42739     .00000   .49081       .00000       .00000     .00000      .00000    .000000E+00
1.42174     .02527   .49472       .35855      -.10527     .00001     -.00008    .776654E-09
1.40475     .05095   .50676       .73072      -.21886     .00003     -.00017    .156074E-08
1.37627     .07746   .52795      1.12821      -.34972     .00004     -.00029    .235120E-08
1.33617     .10531   .56011      1.55718      -.50788     .00005     -.00046    .312468E-08
1.28449     .13511   .60609      2.01001      -.70375     .00006     -.00071    .380901E-08
1.22183     .16761   .66985      2.44656      -.94391     .00006     -.00108    .423490E-08
1.15018     .20377   .75590      2.75124     -1.21376     .00005     -.00157    .407439E-08
1.07461     .24466   .86595      2.66063     -1.42370     .00002     -.00212    .290650E-08
1.00452     .29119   .99102      1.92341     -1.37901     .00000     -.00241    .963548E-09
  .94924     .34356  1.10981       .90421     -1.06006     .00000     -.00219   -.261521E-09
  .91150     .40102  1.20362       .19089      -.69381     .00000     -.00165   -.439089E-09
  .88782     .46232  1.26867      -.12354      -.42292     .00000     -.00113   -.291637E-09
  .87334     .52625  1.31110      -.20899      -.25454     .00000     -.00075   -.170783E-09
  .86444     .59194  1.33824      -.20306      -.15498     .00000     -.00049   -.106839E-09
  .85890     .65878  1.35555      -.16904      -.09579     .00000     -.00033   -.724216E-10
  .85542     .72636  1.36661      -.12961      -.05979     .00000     -.00022   -.502642E-10
  .85322     .79442  1.37364      -.09197      -.03708     .00000     -.00014   -.336329E-10
  .85187     .86280  1.37800      -.05812      -.02173     .00000     -.00008   -.203644E-10
  .85113     .93136  1.38041      -.02795      -.01011     .00000     -.00004   -.950541E-11
  .85089    1.00000  1.38118       .00000       .00000     .00000      .00000    .000000E+00
```

13. RESULTS AND PLANS

* A Fortran program, DROPGEN, has been developed and applied to a problem of extensional flow of a jet of Maxwell model fluid. The program calculates a one dimensional model of the growth of a perturbation of the jet, following the process of droplet formation up to the emergence of a singularity in curvature on the surface of the jet.

* The program is designed to be easily adapted to handle similar problems for other model materials, particularly for materials with hereditary constitutive laws which are best treated in a coordinate system fixed in the material. Such materials ordinarily offer great computational difficulties.

* The program is constructed modularly so that its parts may be abstracted and rearranged for use on other problems of different geometry.

* The program contains a convenient subroutine for five-point central difference approximations to higher derivatives.

* The program contains a subroutine with an original, fast and accurate solution of a set of inhomogeneous linear equations arising from a pentadiagonal matrix of coefficients.

* The calculations for a Maxwell model reveal the development during drop formation of a singularity in the curvature along the axis of the jet which is suggestive of a phenomenon observed by Jones and Ree [1982]. This singularity seems to arise from the elastic response of the filament.

* The effects of changes in relaxation time, modulus, amplitude and form of the perturbation and of different BKZ models of the fluid should be explored with DROPGEN.

* Improvements to DROPGEN might be achieved by rewriting the program to allow adjustment of the time step and/or the node interval in the course of the calculation.

* To extend the computations beyond the time of the appearance of the singularity it would be necessary to interpolate the position of the singularity between nodes and to adjust a force balance there. Markovitch and Renardy [1985] in a similar problem illustrate a technique which might be adapted for this purpose.

* DROPGEN can probably be adapted with a moderate amount of work to treat the problem of the extension under the influence of gravity of a pendant drop of BKZ model liquid.

LITERATURE CITED

Avula, Xavier J. R. [1973]
Application of large elastic deformation theory to the calculation of liquid drop shapes of some polymers, Rheol. Acta 12, 200-205

Bogy, D. B. [1977]
Use of one-dimensional Cosserat theory to study instability in a viscous liquid jet, Phys. Fluids 21, 190-197

Bogy, D. B. [1979]
Drop formation in a circular liquid jet, Ann. Rev. Fluid Mech. 11, 207-228

Bousfield, D. W., R. Keunings, G. Marrucci and M. M. Denn [1984]
Dynamics of filament breakup, extended abstract of 1984 Annual AIChE Meeting, San Francisco, Nov. 25-29

Bousfield, D. W., G. Marrucci and M. M. Denn [1984]
Dynamics of liquid filament breakup, abstract of paper read at the IXth International Congress on Rheology, October 8-13, Mexico City.

Bousfield, D. W., R. Keunings, G. Marrucci and M. M. Denn [1986]
Nonlinear analysis of the surface tension driven breakup of viscoelastic filaments, Journ. Non-Newtonian Fluid Mechanics (to appear)

Carnahan, B., H. A. Luther and J. O. Wilkes [1969]
Applied Numerical Methods, John Wiley & Sons, 441-442

Donnelly, R. J. and W. Glaberson [1966]
Experiments on the capillary instability of a liquid jet, Proc. Roy. Soc. A290, 547-556

Goedde E. F. and M. C. Yuen [1970]
Experiments on liquid jet instabilty, Journ. Fluid Mech. 40, 495-511

Gordon, M., J. Yerushalmi and R. Shinnar [1973]
Instability of jets of non-Newtonian fluids, Trans. Soc. Rheology 17,303-324

Green, A. E. [1976]
On the non-linear behaviour of fluid jets, Int. Journ. Engng. Sci. 14, 49-63

Jones, W. M. and I. J. Rees [1982]
The stringiness of dilute polymer solutions, Journ. Non-Newtonian Fluid Mechanics 11, 257-268

Kearsley, E. A. [1983]
Development of Methods of Evaluating Extensional Properties of Polymer Solutions from Measurements of Shear, Report ARCSL-CR-83076, Chemical Systems Laboratory, Aberdeen Proving Ground, MD 21010, 12.

Kearsley, E. A. [1984]
Rheology of Dissemination, Report CRDC-CR-84097, Chemical Research and Development Center, Aberdeen Proving Ground, MD 21010, 12.

Keunings, R. [1984]
A finite element method for a class of viscoelastic flows in deforming domains applied to jet breakup, abstract of paper read at the IXth International Congress on Rheology, October 8-13,Mexico City.

Keunings, R. [1986]
An algorithm for the simulation of transient viscoelastic flows with free surfaces, Journ. Computational Phys. 62, 199-232

Lafrance, P. [1975]
Nonlinear breakup of a laminar liquid jet, Phys. Fluids 18, 428-432

Markovich, P. and M. Renardy [1985]
A finite difference study of the stretching and break-up of filaments of polymer solution, Jour. Non-Newtonian Fluid Mech. 17, 13-22

Plateau, M. T. [1856]
On the recent theories of the constitution of jets of liquid issuing from circular orifaces, Phil. Mag. 12, 286-297

Rayleigh, Lord [1879]
On the stability of jets, Proc. London Math. Soc. 10, 4-13

Savart, F. [1833]
Memoire sur la constitution des veines liquides lancees par des orifices circulaire en mince paroi, Ann. de Chim. 53, 337-386

Taub, H. H. [1976]
Investigation of nonlinear waves on liquid jets, Phys. Fluids 19, 1124-1129

APPENDIX A

DERIVATIVES CALCULATED FROM A FIVE POINT POLYNOMIAL FIT

Consider a physical quantity represented by a real valued function of a real valued variable, F(Z). If we are given values of this function measured or calculated at intervals along the Z axis, provided that the function is smooth enough and the intervals are small enough, we can approximate the function and its derivatives with respect to Z in terms of these given values. The easiest way to do this is to represent the function by a smooth curve fitted to the given values. In particular, we can approximate derivatives up to the fourth by fitting a fourth order polynomial.

We will restrict our consideration to the simple case when the intervals along the Z axis are regular. We can then represent Z at the given values by an integer, say I, where Z=I Z and Z is the fixed interval between adjacent values. It is convenient also to represent the physical quantity by a real valued function of the integer variable, G(I). To calculate derivatives of F(Z) from derivatives of G(I) is a trivial matter.

In order to approximate the derivatives of the physical quantity at a point corresponding to $I=I_0$, we represent the quantity at this point by a fourth order polynomial taking on the value of G(I) at the five points centered on the point at $I=I_0$. We use the "local" integer variable J given by $I-I_0$ so that J is zero at I_0, the central point, and antisymmetric about it, ranging from -2 to +2 over the five points. Upon designating E(J) as the (unique) fourth order polynomial in J equal to G(I) at the five points we can immediately write down the following expression for it:

$$
\begin{aligned}
E(J) = {} & \frac{(J-1)J(J+1)(J+2)}{24} G(I_0+2) - \frac{(J-2)J(J+1)(J+2)}{6} G(I_0+1) + \\
& + \frac{(J-2)(J-1)(J+1)(J+2)}{4} G(I_0) + \qquad\qquad \text{(A1)} \\
& + \frac{(J-2)(J-1)J(J+1)}{24} G(I_0-2) - \frac{(J-2)(J-1)J(J+2)}{6} G(I_0-1)
\end{aligned}
$$

This polynomial is designed to be equal transparently to G(I) at the five points symmetric about I_0. When the coefficients of the powers of J are collected together the polynomial appears in a form more convenient for differentiation, viz.:

$$E(J) = aJ^4+bJ^3+cJ^2+dJ+e \tag{A2}$$

where the coefficients are as follows:

$$\begin{aligned} 24a &= 6G(I_0) - 4(G(I_0+1)+G(I_0-1)) + G(I_0+2)+G(I_0-2) \\ 12b &= 2(G(I_0-1)-G(I_0+1)) + G(I_0+2)-G(I_0-2) \\ 24c &= 16(G(I_0+1)+G(I_0-1)) - 30G(I_0) - (G(I_0+2)+G(I_0-2)) \\ 12d &= 8(G(I_0+1)-G(I_0-1)) - (G(I_0+2)-G(I_0-2)) \\ e &= G(I_0) \end{aligned} \tag{A3}$$

The derivatives of the physical quantity, F(Z), evaluated at the point $Z=I_0\Delta Z$ are calculated from the derivatives of E(J) at J=0.

$$\begin{aligned} F(Z) &= e \\ F'(Z) &= d/\Delta Z \\ F''(Z) &= 2c/(\Delta Z)^2 \\ F'''(Z) &= 6b/(\Delta Z)^3 \\ F''''(Z) &= 24a/(\Delta Z)^4 \end{aligned} \tag{A4}$$

In applying these equations for the derivatives to our problem we must treat seperately the points where I_0 is equal to 1, 2, N_0-1, and N_0 because some of the necessary values of G(I) are for values of I outside the half period of F(Z) of the calculation and are not directly available. We must use the symmetries and periodicity of our particular problem to find these values. Quantities which are directed axially, such as the displacement of the nodes, have differences which are antisymmetric about the ends of the half period, that is, they conform to the following continuation beyond the ends:

$$G(0)=2G(1)-G(2),\ G(-1)=2G(1)-G(3)$$
$$G(N_0+1)=2G(N_0)-G(N_0-1),\ G(N_0+2)=2G(N_0)-G(N_0-2) \qquad (A5)$$

For scalar and radially directed quantities such as the radius, which have zero slope at the endpoints, the following equations apply:

$$G(0)=G(2),\ G(-1)=G(3)$$
$$G(N_0+1)=G(N_0-1),\ G(N_0+2)=G(N_0-2) \qquad (A6)$$

Equations (A4) can be substituted into equations (A3) (supplemented when necessary with equations (A5) or (A6) according to the symmetries of F(Z)) to evaluate the derivatives at the nodes.

This algorithm is used with the symmetries of equation (A5) in the subroutines DERIV and SLOPES. In the main program the vector H2 containing the axial deviations of the nodes from their initial positions is passed to the subroutine DERIV as the vector G. The derivatives of Z with respect to Z_0 up to the fourth are returned in the matrix E. The addition of one in the calculation of the first derivative takes into account the fact that we calculate from the deviations rather than the positions of the nodes.

The subroutine SLOPES is used in calculating the radial inertia. The velocity at the nodes is passed to vector G of the subroutine as vector V2 from the main program. The first and second spatial derivatives only are returned from the subroutine, also in matrix E.

Blank

APPENDIX B

SOLUTION OF LINEAR EQUATIONS ARISING FROM A PENTADIAGONAL MATRIX

The system of linear equations to be solved consists of N_0 equations in N_0 unknowns, A_i. The constant coefficients of the equations are assigned the values a_i, b_i, c_i, d_i and e_i to give the following scheme:

$$c_1A_1+d_1A_2+e_1A_3 = f_1$$

$$b_2A_1+c_2A_2+d_2A_3+e_2A_4 = f_2$$

$$a_3A_1+b_3A_2+c_3A_3+d_3A_4+e_3A_5 = f_3$$

$$a_4A_2+b_4A_3+c_4A_4+d_4A_5+e_4A_6 = f_4$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

$$a_iA_{i-2}+b_iA_{1-1}+c_iA_i+d_iA_{i+1}+e_iA_{i+2} = f_i \qquad (B1)$$

$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots$$

$$a_{N_0-3}A_{N_0-5}+b_{N_0-3}A_{N_0-4}+c_{N_0-3}A_{N_0-3}+d_{N_0-3}A_{N_0-2}+e_{N_0-3}A_{N_0-1} = f_{N_0-3}$$

$$a_{N_0-2}A_{N_0-4}+b_{N_0-2}A_{N_0-3}+c_{N_0-2}A_{N_0-2}+d_{N_0-2}A_{N_0-1}+e_{N_0-2}A_{N_0} = f_{N_0-2}$$

$$a_{N_0-1}A_{N_0-3}+b_{N_0-1}A_{N_0-2}+c_{N_0-1}A_{N_0-1}+d_{N_0-1}A_{N_0} = f_{N_0-1}$$

$$a_{N_0}\ A_{N_0-2}+b_{N_0}\ A_{N_0-1}+c_{N_0}\ A_{N_0} = f_{N_0}$$

The algorithm we develop for solving this system is a form of Gaussian elimination. It is a generalization of the Fortran program for solving a tridiagonal system given by Carnahan, Luther and Wilkes [1969] in Applied Numerical Methods.

The ith unknown will be given by a recursion relation in the form of a linear combination of the (i+1)th and (i+2)th unknowns, thus:

$$A_i = \gamma_i + \delta_i A_{i+1} + \epsilon_i A_{i+2} \tag{B2}$$

where γ_i ,δ_i and ϵ_i are to be evaluated from the coefficients of the system (B1). With this equation we can express A_{i-1} and A_{i-2} in terms of A_i and A_{i+1}.

$$\begin{aligned} A_{i-1} &= \gamma_{i-1} + \delta_{i-1} A_i + \epsilon_{i-1} A_{i+1} \\ A_{i-2} &= \gamma_{i-2} + \delta_{i-2}\,\gamma_{i-1} + (\epsilon_{i-2} + \delta_{i-1}\delta_{i-2}) A_i + \delta_{i-2}\,\epsilon_{i-1} A_{i+1} \end{aligned} \tag{B3}$$

Upon putting these equations into the ith equation of the system (B1) we can see that the Greek coefficients of equation (B2) also can be evaluated by recursion relations among themselves. For i ranging from 3 to N_0-2 these relations are

$$\begin{aligned} \gamma_i &= (f_i - b_i\,\gamma_{i-1} - a_i(\gamma_{i-2} + \delta_{i-2}\,\gamma_{i-1}))/\Xi_i \\ \delta_i &= -(a_i\,\delta_{i-2}\,\epsilon_{i-1} + b_i\,\epsilon_{i-1} + d_i)/\Xi_i \\ \epsilon_i &= -e_i/\Xi_i \\ \Xi_i &= a_i(\epsilon_{i-2} + \delta_{i-1}\,\delta_{i-2}) + b_i\,\delta_{i-1} + c_i \end{aligned} \tag{B4}$$

These equations may be used as recursion relations because the Greek coefficients on the right are all of order less than i. The Greek coefficients for i=1 and 2 can be directly evaluated from the equations of (B1) as follows:

$$\begin{array}{ll}
\gamma_1=f_1/\Xi_1 & \gamma_2=(c_1f_2-b_2d_1)/\Xi_2 \\
\delta_1=-d_1/\Xi_1 & \delta_2=(b_2e_1-c_1d_2)/\Xi_2 \\
\epsilon_1=-e_1/\Xi_1 & \epsilon_2=-c_1e_2/\Xi_2 \\
\Xi_1=-c_1 & \Xi_2=c_1c_2-b_2d_1
\end{array} \qquad \text{(B5)}$$

This set of equations is still not sufficient to solve the problem because equation (B2) can not directly be used to find the (N_0)th and (N_0-1)th unknowns. However, it can be used to eliminate all other unknowns from the (N_0)th an (N_0-1)th equations of (B1) and the resulting pair of equations can be solved for the two unknowns.

$$A_{N_0-1}=(R_1Q_2-R_2Q_1)/(P_1Q_2-P_2Q_1)$$

$$A_{N_0}=(P_1R_2-R_1P_2)/(P_1Q_2-P_2Q_1)$$

where (B6)

$$\begin{array}{ll}
P_1=b_{N_0}+a_{N_0}\delta_{N_0-2} & P_2=c_{N_0-1}+b_{N_0-1}\delta_{N_0-2}+a_{N_0-1}(\epsilon_{N_0-3}+\delta_{N_0-2}\delta_{N_0-3}) \\
Q_1=c_{N_0}+a_{N_0}\epsilon_{N_0-2} & Q_2=d_{N_0-1}+b_{N_0-1}\epsilon_{N_0-2}+a_{N_0-1}\delta_{N_0-3}\epsilon_{N_0-2} \\
R_1=f_{N_0}-a_{N_0}\gamma_{N_0-2} & R_2=f_{N_0-1}-b_{N_0-1}\gamma_{N_0-2}-a_{N_0-1}(\gamma_{N_0-3}+\delta_{N_0-3}\gamma_{N_0-3})
\end{array}$$

Equations (B2), (B4), (B5) and (B6) can be used to evaluate the unknown A_i's of system (B1) and this set of equations constitutes a complete solution of the pentadiagonal system.

This algorithm is executed in subroutine VBAND. The coefficients of the system of linear equations (B1) are passed to the subroutine in the form of a 6 X N_0 matrix, CMAT. The i'th column of CMAT is formed of the coefficients of the A_j's in the i'th equation augmented by f_i, the right hand side of the equation. The number N_0 is passed to the subroutine as well as five vectors used for storing the the solution A as well as the four intermediate Greek coefficients, gamma, delta, epsilon and xi. The vectors are passed to the subroutine as a strategem to allow dimensioning from the main program.

The algorithm proceeds in two stages; the intermediary Greek coefficients are calculated by use of equations (B5) and subsequently recursion relation (B4); then, the unknown A_i's are calculated starting with the (N_0)th and (N_0-1)th through equation (B6) and then the remaining A_i's in descending order through recursion relation (B2).

APPENDIX C

EXTRAPOLATION OF ACCELERATION, VELOCITY AND DISPLACEMENT

Extrapolations of acceleration, velocity and displacement are needed in calculating the acceleration at each time step. To make these extrapolations we use the following accelerations: AF, an estimate of the acceleration one time step in the future; AP, acceleration at the present time; AR, of the most recent past; AO, of two time steps in the past. The acceleration is taken to be a cubic curve in time passing through these four values. Then, velocity and displacement at one time step in the future are given by the following equations:

$$\begin{gathered} VF=VP+(9AF-5AR+AO+19AP)\Delta t/24 \\ HF=HP+VP\ \ t+(38AF-36AR+7AO+171AP)(\Delta t)^{2}/360 \end{gathered} \tag{1}$$

where HF and VF are the velocity and displacement at one step into the future and HP and VP are are the present velocity and displacement. At the first iteration AF is simply the projection by quadratic curve through the three earlier accelerations and is given by the equation:

$$AF=AO+3(AP-AR) \tag{2}$$

The past times AR and AO are not available in calculating the first time step so that a linear extrapolation of acceleration must be used. In that case the following equations apply:

$$\begin{gathered} VF=VP+(AF+AP)\Delta t/2 \\ HF=HP+VP\ \ t+(2AP+AF)(\Delta t)^{2}/6 \end{gathered} \tag{3}$$

$AF=AP$ for first iteration

For calculating the second time step the acceleration AO is not available and a quadratic curve is used for extrapolation. In this case, we have the following equations:

$$VF=VP+(5AF-AR+AP)\Delta t/12$$

$$HF=HP+VP\ t+(10AP+3AF-AR)(\Delta t)^{2}/24 \qquad (4)$$

$AF=2AP-AR$ for first iteration.

APPENDIX D

CALCULATING INTEGRALS BY UPDATING

For a BKZ material, the viscoelastic force gradient is expressed through integrals over the history of the stretch and its derivatives. In the general case the calculation of these integrals requires that the history of the material be stored for a period into the past determined by the rate of the fading memory. The calculations of this report are done in that way to illustrate this general method but, for special kernal functions of the BKZ model other methods of integration may prove more convenient.

In the example of the Maxwell model, we might have done the integration in way which does not require storing the complete history. The Maxwell model may be expressed in terms of a differential equation rather than an integral equation. In that case, the stress (and thus the force gradient, also) is expressible as a sum of integrals, each of which can be evaluated by updating quantities at each time step.

Consider an integral of the following form:

$$A(t)=\int_0^t G(t)H(\tau)\,d\tau \tag{C1}$$

We can use the trapezoidal rule to approximate this integral and thus, to computational accuracy, we can write:

$$\begin{aligned} A(N\Delta t)=G(N\Delta t)\sum_{n=1}^{N-1} H(n\Delta t)\ \Delta t+ \\ +G(N\Delta t)\,[H(0)+H(N\Delta t)]\ \Delta t/2 \end{aligned} \tag{C2}$$

where $t=N\Delta t$ and $\tau=n\Delta t$. The quantity A is here evaluated at an integral number of time steps from time zero. The integral for the next time step can be worked out by substituting N+1 for N in this equation and expressing the resulting summation in terms of A(N t). This process leads to the following equation:

$$\begin{aligned} A((N+1)\Delta t) = {} & A(N\Delta t)G((N+1)\Delta t)/G(N\Delta t) + \\ & +G((N+1)\Delta t)[H(N\Delta t)+H((N+1)\Delta t)]\,\Delta t/2 \end{aligned} \qquad \text{(C3)}$$

With this equation the integral can be evaluated by updating from a time one interval earlier. Of course, to start off the the method an evaluation of A(0) must be supplied.

END

1-87

DTIC